

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-039752

(43)Date of publication of application : 13.02.1998

(51)Int.Cl.

G09C 1/00
G09C 1/00
G09C 1/00
H04L 9/08
H04L 9/30
H04L 9/32

(21)Application number : 08-189730

(71)Applicant : NIPPON TELEGR & TELEPH CORP
<NTT>

(22)Date of filing : 18.07.1996

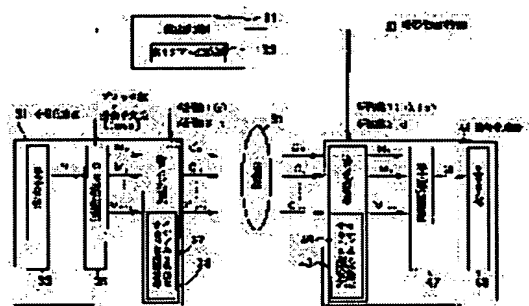
(72)Inventor : TAKAGI TAKESHI
NAITO SHOZO

(54) COMMUNICATION AND CERTIFICATION METHOD BY OPEN KEY CIPHER, AND DEVICE THEREFOR

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a constitution method for an open key ciphering system and device therefor which has a strength of same level or more against a complete deciphering compared with a conventional open key cipher on a rational integer ring, and has a higher strength than ever against a broadcasting attack.

SOLUTION: A key forming device 21 forms prime ideals (p) , (q) in an integer ring (O) on an algebraic number field for making them as a first secret key, and makes the remainders of their product $(n)=(p)(q)$ as a first open key. Further, a second secret key d and a second open key e are formed from (p) and (q) . A ciphering device 31 divides an inputted declarative sentence M into blocks, and ciphers them by performing a modulo ideal (n) raising operation to e th power, and outputs ciphered sentences $(C_0, C_1, \dots, C_{r-1})$ to a communication path 51. A decoding device 41 decodes the inputted blocks of the ciphered sentences by performing a modulo ideal (n) raising operation to d th power, and corporates the decoded blocks of the declarative sentence for outputting the declarative sentence.



LEGAL STATUS

[Date of request for examination] 26.12.2000

[Date of sending the examiner's decision of rejection] 08.04.2003

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

Best Available Copy

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(10) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-39752

(43) 公開日 平成10年(1998) 2月18日

(51) Int. Cl.	識別記号	庁内整理番号	F I	法務表示箇所
G 0 9 C 1/00	6 3 0	7259-5J	G 0 9 C 1/00	8 3 0 Z
		7259-5J		8 3 0 F
	6 2 0	7259-5J		8 2 0 B
	6 4 0	7259-5J		8 4 0 B
H 0 4 L 9/00			H 0 4 L 9/00	8 0 1 Z

審査請求 未請求 請求項の数53 OL (全 35 頁) 最終頁に続く

(21) 出願番号 特願平8-189730

(22) 出願日 平成8年(1996) 7月18日

(71) 出願人 000004226

日本電信電話株式会社

東京都新宿区西新宿三丁目19番2号

(72) 発明者 高木 剛

東京都新宿区西新宿三丁目19番2号 日本

電信電話株式会社内

(73) 発明者 内藤 昭三

東京都新宿区西新宿三丁目19番2号 日本

電信電話株式会社内

(74) 代理人 弁理士 三好 秀和 (外1名)

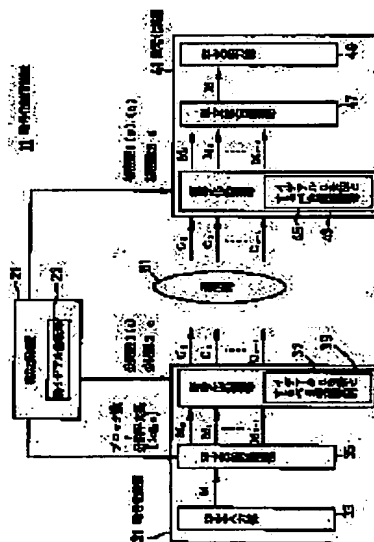
(54) 【発明の名称】 公開鍵暗号による通信および解読方法、ならびにそれらの装置

(57) 【要約】

【課題】 従来の有理整数環上の公開鍵暗号と比較して、完全解読に対しては同程度以上の強度を持ち、同報通信攻撃に対して従来より高い強度を持つ公開鍵暗号方式および装置の構成法を提供する。

【解決手段】 鍵生成装置2は、代数体上の整数環Oにおける素イデアル(p)、(a)を生成して第1の秘密鍵とし、その積(n)=(p)・(a)の剰余類を第1の公開鍵とする。また(p)、(a)から第2の秘密鍵dと第2の公開鍵eを生成する。暗号化装置3は、入力された平文Mをブロックに分割し、イデアル(n)を法とするe乗演算により暗号化を行い、暗号文(C0, C1, ..., Cr-1)を通信路5.1に出力する。復号化装置4

1は、入力された暗号文のブロックに対しイデアル(n)を法とするd乗演算により復号化を行い、復号化された平文ブロックを統合して、平文Mを出力する。



【特許請求の範囲】

【請求項1】 ある素数データに基づいて、代数体上の整数環における素イデアルであることを満たすような互いに異なる二つの自然数データの組である第1の秘密鍵データ、ならびに前記自然数データの積である第1の公開鍵データを生成することを特徴とする公開鍵暗号の鍵生成方法。

【請求項2】 ある素数データに基づいて、代数体上の整数環における素イデアルであることを満たすような互いに異なる二つの自然数データの組である第1の秘密鍵データ、ならびに前記自然数データの積である第1の公開鍵データを生成し、

前記素数データのオイラー関数値の計算と、前記第1の秘密鍵データに関する最小公倍数演算とを用いて、ともに自然数データである第2の秘密鍵データと第2の公開鍵データとを生成することを特徴とする公開鍵暗号の鍵生成方法。

【請求項3】 ある素数べきデータである m_0 のオイラー関数値を求めて、

前記オイラー関数値をべき乗した結果に対して m_0 を法とする剰余を計算すると1となり、かつ、前記オイラー関数値より小さいいずれの自然数をべき乗した結果に対して m_0 を法とする剰余を計算しても、1にならない条件を満たすような、互いに異なる二つの自然数データである p_0 と q_0 とを探索し、

p_0 と q_0 との組を第1の秘密鍵データとして出力し、 p_0 と q_0 を乗じてデータ N_0 を求め、 N_0 を第1の公開鍵データとして出力することを特徴とする公開鍵暗号の鍵生成方法。

【請求項4】 前記 p_0 と q_0 とにそれぞれ前記オイラー関数値をべき乗し1を減じた値同士の最小公倍数の値である L_0 を算出し、 L_0 を法とする剰余が1となるデータ値を積とするような二つの自然数データを探索し、

前記探索された二つの自然数データの一方を第2の秘密鍵データ d_0 として、他方を第2の公開鍵データ e_0 として、それぞれ出力することを特徴とする請求項3に記載の公開鍵暗号の鍵生成方法。

【請求項5】 前記 p_0 と q_0 との探索は、ユークリッド的な m_0 次元分体において、0でないイデアルによる、 m_0 に対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して行うことを特徴とする請求項3または4に記載の公開鍵暗号の鍵生成方法。

【請求項6】 前記 p_0 と q_0 との探索は、任意の円分体において、惰性する素数の積で表されるイデアルによる、 m_0 に対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを特徴とする請求項3または4に記載の公開鍵暗

号の鍵生成方法。

【請求項7】 二乗因子を含まないある有理整数データ値である m_s に対して、4を法とする剰余を計算し、この剰余の計算値が1となる場合は、判別式データの値を m_s とし、

この剰余の計算値が1以外の場合は、判別式データの値を $4m_s$ とし、

有理素数のなかから、この有理素数を法として前記判別式データを平方非剰余とするような互いに異なる二つの有理素数である p_s と q_s を探索し、

p_s と q_s の組を、第1の秘密鍵データとして出力し、

p_s と q_s を乗じてデータ N_s を求め、

N_s を第1の公開鍵データとして出力することを特徴とする公開鍵暗号の鍵生成方法。

【請求項8】 前記 p_s と q_s のそれぞれの2乗から1を減じた値同士の最小公倍数データ値である L_s を算出し、

この L_s を法とする剰余が1となるデータ値を積とするような二つの自然数データを探索し、

この二つの自然数データの一方を第2の秘密鍵データ d_s とし、他方を第2の公開鍵データ e_s として、それぞれ出力することを特徴とする請求項7に記載の公開鍵暗号の鍵生成方法。

【請求項9】 前記の p_s と q_s の探索は、

ユークリッド的な2次元において、0でないイデアルによる2次の剰余類の完全代表系である2次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを特徴とする請求項7または請求項8に記載の公開鍵暗号の鍵生成方法。

【請求項10】 ある素数べきデータである m_0 を入力する入力手段と、

m_0 のオイラー関数値を求めるオイラー関数計算手段と、

前記オイラー関数値をべき乗した結果に対して m_0 を法とする剰余を計算すると1となり、かつ、前記オイラー関数値より小さいいずれの自然数をべき乗した結果に対して m_0 を法とする剰余を計算しても、1にならない条件を満たすような、互いに異なる二つの自然数データである p_0 と q_0 とを探索する第1の探索手段と、

p_0 と q_0 を乗じてデータ N_0 を求める乗算手段と、

p_0 と q_0 とにそれぞれ前記オイラー関数値をべき乗し1を減じた値同士の最小公倍数データ値である L_0 を算出し、 L_0 を法とする剰余が1となるデータ値を積とするような二つの自然数データ d_0 および e_0 を探索する第2の探索手段と、

p_0 と q_0 との組を第1の秘密鍵データとして、 N_0 を第1の公開鍵データとして、 d_0 を第2の秘密鍵データとして、 e_0 を第2の公開鍵データとして、それぞれ出力する出力手段と、

を具備したことを特徴とする公開鍵暗号の鍵生成装置。

【請求項1.1】 二乗因子を含まないある有理整数データ値である ms を入力する入力手段と、
 ms に対して、4を法とする剰余を計算し、この剰余の計算値が1となる場合は、判別式データの値を ms とし、この剰余の計算値が1以外の場合は、判別式データの値を $4ms$ とする判別式データ設定手段と、
 有理素数のなかから、この有理素数を法として前記判別式データを平方非剰余とするような互いに異なる二つの有理素数である ps と qs とを探索する第1の探索手段と、
 ps と qs を乗じてデータ Ns を求める乗算手段と、
 ps と qs のそれぞれの2乗から1を引いた値同士の最小公倍数データ値である ls を算出し、この ls を法とする剰余が1となるデータ値を積とするような二つの自然数データ ds および es を探索する第2の探索手段と、
 ps と qs の組を第1の秘密鍵データとして、 Ns を第1の公開鍵データとして、 ds を第2の秘密鍵データとして、 es を第2の公開鍵データとして、それぞれ出力する出力手段と、
 を具備したことを特徴とする公開鍵暗号の鍵生成装置、
 【請求項1.2】 第1および第2の公開鍵により平文データを暗号化して暗号文データを生成する公開鍵暗号による暗号化方法であって、
 平文データを複数ブロックに分割して得られた平文ブロックデータに対して、第2の公開鍵データ値である eo または es をべき乗した値を計算し、
 この計算値に対して、第1の公開鍵であるイデアル No または Ns を法とした剰余を求めることにより、暗号文データを生成することを特徴とする公開鍵暗号による暗号化方法、
 【請求項1.3】 前記イデアルを法とする剰余類の取り方は、
 ユークリッド的な mo 次元分体において、0でないイデアル No を法とする剰余類の完全代表系として、 mo に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とし、
 前記平文データが、それぞれ前記 mo に対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを特徴とする請求項1.2に記載の公開鍵暗号による暗号化方法、
 【請求項1.4】 前記イデアルを法とする剰余類の取り方は、
 ユークリッド的な2次元分体において、0でないイデアル Ns を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とし、
 前記平文データが、それぞれ2つのブロックからなる平文ブロックデータに分割された後に暗号化されることを特徴とする請求項1.2に記載の公開鍵暗号による暗号化

方法、
 【請求項1.5】 平文ブロックデータに対して、第2の公開鍵データ値である eo をべき乗した値を計算するべき乗計算手段と、
 この計算値に対して、第1の公開鍵であるイデアル No を法とした剰余を求める剰余類計算手段と、
 を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な mo 次元分体において、0でないイデアル No を法とする剰余類の完全代表系として、 mo に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号による暗号化装置、
 【請求項1.6】 平文ブロックデータに対して、第2の公開鍵データ値である es をべき乗した値を計算するべき乗計算手段と、
 この計算値に対して、第1の公開鍵であるイデアル Ns を法とした剰余を求める剰余類計算手段と、
 を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な2次元分体において、0でないイデアル Ns を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを特徴とする公開鍵暗号による暗号化装置、
 【請求項1.7】 請求項1.2ないし請求項1.4のいずれか1項記載の暗号化方法により生成された暗号文ブロックデータに対して復号化を施して平文データを生成する公開鍵暗号の復号化方法であって、
 前記暗号文ブロックデータに対して、第2の秘密鍵データ値である do または ds をべき乗した値を計算し、
 この計算値に対して、第1の公開鍵であるイデアル No または Ns を法とした剰余を求めることにより、平文データを生成することを特徴とする公開鍵暗号の復号化方法、
 【請求項1.8】 前記イデアルを法とする剰余類の取り方は、
 ユークリッド的な mo 次元分体において、0でないイデアル No を法とする剰余類の完全代表系として、 mo に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする請求項1.7に記載の公開鍵暗号の復号化方法、
 【請求項1.9】 前記イデアルを法とする剰余類の取り方は、
 ユークリッド的な2次元分体において、0でないイデアル Ns を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを特徴とする請求項1.7に記載の公開鍵暗号の復号化方法、
 【請求項2.0】 暗号文ブロックデータに対して、第2

の秘密鍵データ値である d_0 をべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第1の公開鍵であるイデアル N_0 を法とした剰余を求める剰余類計算手段と、
を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な m_0 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m_0 に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号の復号化装置。

【請求項2.1】 暗号文ブロックデータに対して、第2の秘密鍵データ値である d_s をべき乗した値を計算するべき乗計算手段と、
この計算値に対して、第1の公開鍵であるイデアル N_s を法とした剰余を求める剰余類計算手段と、
を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な2次元分体において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする公開鍵暗号の復号化装置。

【請求項2.2】 第1の公開鍵および第2の秘密鍵により、認証文から暗号化された認証子データを生成する公開鍵暗号方式による認証文生成方法であって、
前記認証文をハッシュ化し、複数ブロックに分割した認証子ブロックデータを生成し、
この認証子ブロックデータに対して、第2の秘密鍵データ値である d_0 または d_s をべき乗した値を計算し、
この計算値に対して、第1の公開鍵データであるイデアル N_0 または N_s を法とした剰余を求めることによって、暗号化認証子データを生成することを特徴とする公開鍵暗号方式による認証文生成方法。

【請求項2.3】 前記イデアルを法とする剰余類の取り方は、
ユークリッド的な m_0 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m_0 に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とし、
前記平文データが、それぞれ前記 m_0 に対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを特徴とする請求項2.2に記載の公開鍵暗号方式による認証文生成方法。

【請求項2.4】 前記イデアルを法とする剰余類の取り方は、
ユークリッド的な2次元分体において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とし、

前記平文データが、それぞれ2つのブロックからなる平

文ブロックデータに分割された後に暗号化されることを特徴とする請求項2.2に記載の公開鍵暗号方式による認証文生成方法。

【請求項2.5】 入力された認証文をハッシュ化した後に分割し、それぞれ m_0 に対するオイラー関数値に等しいブロック数からなる認証子ブロックデータを生成する認証子データ生成手段と、

この生成された認証子ブロックデータに対して、第2の秘密鍵データ値である d_0 をべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第1の公開鍵データであるイデアル N_0 を法とした剰余を求める剰余類計算手段と、
を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な m_0 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m_0 に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文生成装置。

【請求項2.6】 入力された認証文をハッシュ化した後に分割し、それぞれ2つのブロックからなる認証子ブロックデータを生成する認証子データ生成手段と、
この生成された認証子ブロックデータに対して、第2の秘密鍵データ値である d_s をべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第1の公開鍵データであるイデアル N_s を法とした剰余を求めて暗号化認証子データを得る剰余類計算手段と、

を備えてなり、前記剰余類計算手段におけるイデアル N_s を法とする剰余類の取り方は、ユークリッド的な2次元分体において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形体の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文生成装置。

【請求項2.7】 請求項2.2ないし請求項2.4のいずれか1項記載の認証文生成方法により生成された暗号化認証子データを復号化し、平文でありこの暗号化認証子データに相当する認証文データの正当性を検証する認証文検証方法であって、

暗号化された認証子ブロックデータに対して、第2の公開鍵データ値である e_0 または e_s をべき乗した値を計算し、

この計算値に対して、第1の公開鍵であるイデアル N_0 または N_s を法とした剰余を求めることによって、認証子データを復号化し、

この復号化した認証子データと、前記平文の認証子データとが一致していた場合は、認証ないし検証過程を正当と判定し、

この復号化した認証子データと、前記平文の認証子データとが不一致であった場合は、認証ないし検証過程を正

当てない」と判定する。

ことを特徴とする公開鍵暗号方式による認証文検証方法。

【請求項28】 前記イデアルを法とする剰余類の取り方は、

ユークリッド的な m 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする請求項27に記載の公開鍵暗号方式による認証文検証方法。

【請求項29】 前記イデアルを法とする剰余類の取り方は、

ユークリッド的な2次元分体において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを特徴とする請求項27に記載の公開鍵暗号方式による認証文検証方法。

【請求項30】 暗号化された認証子ブロックデータに対して、第2の公開鍵データ値である e_0 をべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第1の公開鍵であるイデアル N_0 を法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、

この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、

を備える公開鍵暗号方式による認証文検証装置であって、

前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、

ユークリッド的な m 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文検証装置。

【請求項31】 暗号化された認証子ブロックデータに対して、第2の公開鍵データ値である e_s をべき乗した値を計算するべき乗計算手段と、

この計算値に対して、第1の公開鍵であるイデアル N_s を法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、

この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、

を備える公開鍵暗号方式による認証文検証装置であって、

前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、

ユークリッド的な2次元分体において、0でないイデアル N

s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを特徴とする公開鍵暗号方式による認証文検証装置。

【請求項32】 前記剰余類計算手段における剰余の計算は、前記 m に対するオイラー関数値に等しい数の各成分毎に行うことを特徴とする請求項15、請求項20、請求項25、または請求項30のいずれか1項に記載の装置。

【請求項33】 前記剰余類計算手段における剰余の計算は、2つの各成分毎に行うことを特徴とする請求項16、請求項21、請求項26、または請求項31のいずれか1項に記載の装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、暗号化鍵が公開され、復号化鍵のみが秘匿される公開鍵方式の暗号に係り、特に、従来のリベスト・シャミール・アドルマン暗号（以下、R.S.A暗号と略す）より同報通信攻撃に対する強度を高めた公開鍵暗号による通信および認証方式並びにそれらの装置に関する。

【0002】

【従来の技術】電気通信において、通信内容の盗聴を防ぎ、改竄を検出するために暗号技術は不可欠である。特に、鍵管理が簡便な公開鍵暗号が有効であり、広く利用されつつある。代表的な公開鍵暗号のアルゴリズムとして、べき（冪）乗剰余演算を用いるR.S.A暗号があり、既に実用化されている。

【0003】

以下、池野、小山による「現代暗号理論」（電子情報通信学会発行）に基づいて、R.S.A暗号の基本原理解、その鍵生成法および認証方法について説明する。

【R.S.A暗号の基本原理解】暗号化鍵は (e, N) の組であり、対応する復号化鍵は (d, N) の組である。

【0004】 e と N とは公開鍵であり、 d は秘密鍵である。平文を M 、暗号文を C とする。暗号化 E と復号化 D のアルゴリズムは、次の式（1）および式（2）で表される。

【数1】

$$C = E(M) = M^e \bmod N \quad \dots (1)$$

$$M = D(C) = C^d \bmod N \quad \dots (2)$$

但し、 M と C とは0から $N-1$ の間の整数である。もし元のメッセージが N より大きければ、サイズ N のブロックに分割して逐一、暗号化・復号化を適用すればよい。

【0005】暗号化と復号化は、1対1かつ上への写像である。従って、 M と C を代表して M で表すと、

【数2】

$$D(E(M)) = E(D(M)) = M \quad \dots (3)$$

式（3）が成立する。具体的には、

【数3】

$$Med \equiv M \pmod{N} \quad \dots (4)$$

式(4)が成立する。但し式(4)および以下の記述において、「 \equiv 」は合同を示す記号とする。式(4)がすべてのMに対して成立するような、暗号鍵e、d、Nの生成手順は以下のとおりである。

【0006】(RSA暗号の鍵生成) まず、任意の相異なる二つの大きな素数p、qを選び、その積 $N = p \cdot q$ を計算する(第1段階)。

【0007】次いで、 $(p-1)$ と $(q-1)$ の最小公倍数Lを計算し、Lと互いに素でより小さな任意の整数eを選ぶ(第2段階)。

【0008】

【数4】

$$L = \text{LCM}((p-1), (q-1)) \quad \dots (5)$$

$$\text{GCD}(e, L) = 1, \quad 1 < e < L \quad \dots (6)$$

次いで、第2段階で求めたeとLをもとに、次の合同式(7)を解き、dを求める(第3段階)。

【0009】

【数5】

$$e \cdot d \equiv 1 \pmod{L} \quad \dots (7)$$

式(7)からdを求めるには、ユークリッドの互除法を用いればよい。また、前記の生成手順は、まず、第2段階でeを先に選び、第3段階でdを計算しているが、逆にdを先に選び、後でeを計算してもよい。

【0010】このような、従来のRSA型公開鍵による暗号化通信装置の構成例を図33に示す。

【0011】(RSA暗号による認証) RSA暗号による認証通信は、以下のように行われる。まず、送信者は、認証文Mをハッシュ関数hによりハッシュ化し、認証子h(M)を得る。次いで認証子h(M)を自己の秘密鍵dで暗号化し、暗号化認証子h(C)を得る。次いで、暗号化認証子h(C)と認証文Mの組を送信者から受信者に送る。

【0012】

【数6】

$$h(C) \equiv (h(M))^d \pmod{N} \quad \dots (8)$$

受信者は、暗号化認証子h(C)と認証文Mの組を受信すると、送信者の公開鍵eを使用して、暗号化認証子h(C)を復号化し認証子h(M)を得る。

【0013】

【数7】

$$h(M) \equiv (h(C))^e \pmod{N} \quad \dots (9)$$

次いで、受信した認証文Mをハッシュ関数hによりハッシュ化し、認証子h(M)を得る。そして復号化した認証子h(M)とハッシュ化した認証子h(M)とを比較して正当性を判断する。すなわち、両者が一致していれば認証文が正当であり、不一致であれば不当と判断する。

【0014】このように、RSA型の公開鍵暗号装置は、公開鍵と秘密鍵の演算を逆に適用することにより、

認証通信装置としても使用することができる。

【0015】なお、この認証通信において、前記暗号化認証子h(C)と認証文Mの組をさらに受信者の公開鍵を用いて暗号化する暗号化認証通信も、認証通信と暗号化通信とを組み合わせて行うことによって可能であることはいうまでもない。

【0016】ところで、一般に暗号技術の性能評価の尺度は、暗号を解読しようとする攻撃に対する安全性の強度と、暗号化・復号化の速度である。強度が高く、速度の速い暗号が優秀な暗号である。

【0017】RSA暗号などの公開鍵暗号は、暗号化鍵である公開された公開鍵から、復号化鍵である秘密鍵を得ることが計算的に困難であることに安全性の根拠を置いている。

【0018】RSA暗号では、公開鍵が素因数分解できれば、暗号文から平文を得ることができるが、文献「R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Comm. ACM, vol. 21, No. 2, pp. 120-126, (1978)」に示されている。これらの安全性の評価は、完全解読を対象としたものであり、この場合には公開鍵の素因数分解の十分性あるいは計算的な同値性が示されている。

【0019】

【発明が解決しようとする課題】しかしながら、J. Hastadは、文献「J. Hastad, "On using RSA with low exponent in a public-key network", Proceeding of CRYPTO'85, Springer-Verlag, (1986)」において、同報通信攻撃と呼ばれるRSAタイプの公開鍵暗号に対する解読法を提案した。

【0020】これにより、RSAタイプの公開鍵暗号では、同一の発信者から複数の受信者に対して、同一内容の平文をそれぞれ異なる公開鍵により暗号化して送信された同報通信が盗聴されると、複数の暗号文から公開鍵を素因数分解することなく平文を得ることができることがわかり、安全性の評価基準を見直す必要がでてきた。

【0021】RSA暗号では暗号化のべき指数を大きくすることにより同報通信攻撃を回避できるが、べき指数を大きくするとべき乗計算時間および剰余計算時間が増加し、暗号化の速度が遅くなるという問題点があった。

【0022】このオリジナルなRSA暗号の同報通信攻撃に対する防御を強化した暗号として、楕円曲線を用いたRSA暗号が知られている。この楕円曲線を用いたRSA暗号は、文献「森門、秀典、小山、謙二、"3次曲線上のRSA型暗号方式の同報通信における安全性", 信学技報ISEC94-10, (1994)」において、その同報通信攻撃に対する安全強度が評価されているが、暗号化および復号化の速度がオリジナルのRSA暗号と比較して5倍以上遅くなり、用途が限定されるという問題点があった。

【0023】以上の問題点に鑑み、本発明の第1の課題は、従来の有理整数環上のRSA暗号と比較して同程度以上の強度を持ち、同報通信攻撃に対して従来以上の強い暗号方式および装置の構成法を与えることである。

【0024】また、本発明の第2の課題は、楕円曲線上のRSA暗号と比較して、暗号化および復号化処理が高速である暗号方式および装置の構成法を与えることである。

【0025】また、本発明の第3の課題は、認証装置としても利用可能であり、ひとつの装置で暗号通信と認証通信の両用が可能となるような暗号および認証方式および装置の構成法を与えることである。

【0026】

【課題を解決するための手段】上記課題を解決するため、本発明においては、従来の有理整数環上でのRSA暗号を代数体の整数環の上へ拡張した公開鍵暗号方式の具体的な構成方法である、代数体の整数環上のイデアルを用いた鍵生成方法、べき乗演算方法、およびイデアルを法とする剰余類の演算方法を与え、さらに鍵生成装置、暗号通信装置および認証通信装置として実現する。

【0027】すなわち、請求項1記載の発明は、ある素数データに基づいて、代数体上の整数環における素イデアルであることを満たすような互いに異なる二つの自然数データの組である第1の秘密鍵データ、ならびに前記二自然数データの積である第1の公開鍵データを生成することを要旨とする公開鍵暗号の鍵生成方法である。

【0028】また、請求項2記載の発明は、ある素数データに基づいて、代数体上の整数環における素イデアルであることを満たすような互いに異なる二つの自然数データの組である第1の秘密鍵データ、ならびに前記二自然数データの積である第1の公開鍵データを生成し、前記素数データのオイラー関数値の計算と、前記第1の秘密鍵データに関する最小公倍数演算とを用いて、ともに自然数データである第2の秘密鍵データと第2の公開鍵データを生成することを要旨とする公開鍵暗号の鍵生成方法である。

【0029】また、請求項3記載の発明は、ある素数べきデータである m_0 のオイラー関数値を求めて、前記オイラー関数値をべき乗した結果に対して m_0 を法とする剰余を計算すると1となり、かつ、前記オイラー関数値より小さい自然数をべき乗した結果に対して m_0 を法とする剰余を計算しても、1にならない条件を満たすような、互いに異なる二つの自然数データである p_0 と q_0 とを探索し、 p_0 と q_0 との組を第1の秘密鍵データとして出力し、 p_0 と q_0 を乗じてデータ N_0 を求め、 N_0 を第1の公開鍵データとして出力することを要旨とする公開鍵暗号の鍵生成方法である。

【0030】また、請求項4記載の発明は、請求項3に記載の公開鍵暗号の鍵生成方法において、前記 p_0 と q_0 とにそれぞれ前記オイラー関数値をべき乗し1を減じ

た値同士の最小公倍数の値である L_0 を算出し、 L_0 を法とする剰余が1となるデータ値を積とするような二つの自然数データを探索し、前記探索された二つの自然数データの一つを第2の秘密鍵データ d_0 として、他方を第2の公開鍵データ e_0 として、それぞれ出力することを要旨とする。

【0031】また、請求項5記載の発明は、請求項3または4に記載の公開鍵暗号の鍵生成方法において、前記 p_0 と q_0 との探索は、ユークリッド的な m_0 次元分体において、0でないイデアルによる、 m_0 に対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して行うことを要旨とする。

【0032】また、請求項6記載の発明は、請求項3または4に記載の公開鍵暗号の鍵生成方法において、前記 p_0 と q_0 との探索は、任意の円分体において、特性する素数の積で表されるイデアルによる、 m_0 に対するオイラー関数の値を次数に持つ剰余類の完全代表系上の多次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを要旨とする。

【0033】また、請求項7記載の発明は、二素因子を含まないある有理整数データ値である m_s に対して、4を法とする剰余を計算し、この剰余の計算値が1となる場合は、判別式データの値を m_s とし、この剰余の計算値が1以外の場合は、判別式データの値を $4m_s$ とし、有理整数のなかから、この有理整数を法として前記判別式データを平方非剰余とするような互いに異なる二つの数である p_s と q_s を探索し、 p_s と q_s の組を第1の秘密鍵データとして出力し、 p_s と q_s を乗じてデータ N_s を求め、 N_s を第1の公開鍵データとして出力することを要旨とする公開鍵暗号の鍵生成方法である。

【0034】また、請求項8記載の発明は、請求項7に記載の公開鍵暗号の鍵生成方法において、前記 p_s と q_s のそれぞれの2乗から1を減じた値同士の最小公倍数データ値である L_s を算出し、この L_s を法とする剰余が1となるデータ値を積とするような二つの自然数データを探索し、この二つの自然数データの一つを第2の秘密鍵データ d_s とし、他方を第2の公開鍵データ e_s として、それぞれ出力することを要旨とする。

【0035】また、請求項9記載の発明は、請求項7または請求項8に記載の公開鍵暗号の鍵生成方法において、前記 p_s と q_s の探索は、ユークリッド的な2次元分体において、0でないイデアルによる2次の剰余類の完全代表系である2次元ベクトルの原点からの距離である、ノルムを使用して求めることによって行うことを要旨とする。

【0036】また、請求項10記載の発明は、ある素数べきデータである m_0 を入力する入力手段と、 m_0 のオイラー関数値を求めるオイラー関数計算手段と、前記オイラー関数値をべき乗した結果に対して m_0 を法とする

剰余を計算すると1となり、かつ、前記オイラー関数値より小さいいずれの自然数をべき乗した結果に対して m_0 を法とする剰余を計算しても、1にならない条件を満たすような互いに異なる二つの自然数データである p_0 と q_0 とを探索する第1の探索手段と、 p_0 と q_0 を乗じてデータ N_0 を求める乗算手段と、 p_0 と q_0 とにそれぞれ前記オイラー関数値をべき乗し1を減じた値同士で最小公倍数データ値である L_0 を算出し、 L_0 を法とする剰余が1となるデータ値を積とするような二つの自然数データ d_0 および e_0 を探索する第2の探索手段と、 p_0 と q_0 の組を第1の秘密鍵データとして、 N_0 を第1の公開鍵データとして、 d_0 を第2の秘密鍵データとして、 e_0 を第2の公開鍵データとして、それぞれ出力する出力手段と、を具備したことを要旨とする公開鍵暗号の鍵生成装置である。

【0037】また、請求項1記載の発明は、二乗因子を含まないある有理整数データ値である m_s を入力する入力手段と、 m_s に対して、4を法とする剰余を計算し、この剰余の計算値が1となる場合は、判別式データの値を m_s とし、この剰余の計算値が1以外の場合は、判別式データの値を $4m_s$ とする判別式データ設定手段と、有理素数のなから、この有理素数を法として前記判別式データを平方非剰余とするような互いに異なる二つの有理素数である p_s と q_s とを探索する第1の探索手段と、 p_s と q_s を乗じてデータ N_s を求める乗算手段と、 p_s と q_s のそれぞれの2乗から1を減じた値同士で最小公倍数データ値である L_s を算出し、この L_s を法とする剰余が1となるデータ値を積とするような二つの自然数データ d_s および e_s を探索する第2の探索手段と、 p_s と q_s の組を第1の秘密鍵データとして、 N_s を第1の公開鍵データとして、 d_s を第2の秘密鍵データとして、 e_s を第2の公開鍵データとして、それぞれ出力する出力手段と、を具備したことを要旨とする公開鍵暗号の鍵生成装置。

【0038】また、請求項1記載の発明は、第1および第2の公開鍵により平文データを暗号化して暗号文データを生成する公開鍵暗号による暗号化方法であって、平文データを複数ブロックに分割して得られた平文ブロックデータに対して、第2の公開鍵データ値である e_0 または e_s をべき乗した値を計算し、この計算値に対して、第1の公開鍵であるイデアル N_0 または N_s を法とした剰余を求めることによって、暗号文データを生成することを要旨とする公開鍵暗号による暗号化方法である。

【0039】また、請求項1記載の発明は、請求項12に記載の公開鍵暗号による暗号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な m_0 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m_0 に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行

四辺体の内部と境界上の格子点とし、前記平文データが、それぞれ前記 m_0 に対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【0040】また、請求項14記載の発明は、請求項12に記載の公開鍵暗号による暗号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な2次元分体において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とし、前記平文データが、それぞれ2つのブロックからなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【0041】また、請求項15記載の発明は、平文ブロックデータに対して、第2の公開鍵データ値である e_0 をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵であるイデアル N_0 を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な m_0 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m_0 に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による暗号化装置である。

【0042】また、請求項16記載の発明は、平文ブロックデータに対して、第2の公開鍵データ値である e_s をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵であるイデアル N_s を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な2次元分体において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを要旨とする公開鍵暗号による暗号化装置である。

【0043】また、請求項17記載の発明は、請求項12ないし請求項14のいずれか1項記載の暗号化方法により生成された暗号文ブロックデータに対して復号化を施して平文データを生成する公開鍵暗号の復号化方法であって、前記暗号文ブロックデータに対して、第2の秘密鍵データ値である d_0 または d_s をべき乗した値を計算し、この計算値に対して、第1の公開鍵であるイデアル N_0 または N_s を法とした剰余を求めることによって、平文データを生成することを要旨とする公開鍵暗号の復号化方法である。

【0044】また、請求項18記載の発明は、請求項17に記載の公開鍵暗号の復号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な m_0 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m_0 に対するオイラー関数

の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする。

【0045】また、請求項19記載の発明は、請求項17に記載の公開鍵暗号の復号化方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な2次元において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを要旨とする。

【0046】また、請求項20記載の発明は、暗号文ブロックデータに対して、第2の秘密鍵データ値である d_o をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵であるイデアル N_o を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な m_o 次元分体において、0でないイデアル N_o を法とする剰余類の完全代表系として、 m_o に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号の復号化装置である。

【0047】また、請求項21記載の発明は、暗号文ブロックデータに対して、第2の秘密鍵データ値である d_s をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵であるイデアル N_s を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な2次元において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを要旨とする公開鍵暗号の復号化装置である。

【0048】また、請求項22記載の発明は、第1の公開鍵および第2の秘密鍵により、認証文から暗号化された認証子データを生成する公開鍵暗号方式による認証文生成方法であって、前記認証文をハッシュ化し、楕円ブロックに分割した認証子ブロックデータを生成し、この認証子ブロックデータに対して、第2の秘密鍵データ値である d_o または d_s をべき乗した値を計算し、この計算値に対して、第1の公開鍵データであるイデアル N_o または N_s を法とした剰余を求めることによって、暗号化認証子データを生成することを要旨とする公開鍵暗号方式による認証文生成方法である。

【0049】また、請求項23記載の発明は、請求項22に記載の公開鍵暗号による認証文生成方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な m_o 次元分体において、0でないイデアル N_o を法とする剰余類の完全代表系として、 m_o に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とし、前記平文デー

タが、それぞれ前記 m_o に対するオイラー関数値に等しいブロック数からなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【0050】また、請求項24記載の発明は、請求項22に記載の公開鍵暗号による認証文生成方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な2次元において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とし、前記平文データが、それぞれ2つのブロックからなる平文ブロックデータに分割された後に暗号化されることを要旨とする。

【0051】また、請求項25記載の発明は、入力された認証文をハッシュ化した後に分割し、それぞれ m_o に対するオイラー関数値に等しいブロック数からなる認証子ブロックデータを生成する認証子データ生成手段と、この生成された認証子ブロックデータに対して、第2の秘密鍵データ値である d_o をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵データであるイデアル N_o を法とした剰余を求める剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアルを法とする剰余類の取り方は、ユークリッド的な m_o 次元分体において、0でないイデアル N_o を法とする剰余類の完全代表系として、 m_o に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による認証文生成装置である。

【0052】また、請求項26記載の発明は、入力された認証文をハッシュ化した後に分割し、それぞれ2つのブロックからなる認証子ブロックデータを生成する認証子データ生成手段と、この生成された認証子ブロックデータに対して、第2の秘密鍵データ値である d_s をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵データであるイデアル N_s を法とした剰余を求めて暗号化認証子データを得る剰余類計算手段と、を備えてなり、前記剰余類計算手段におけるイデアル N_s を法とする剰余類の取り方は、ユークリッド的な2次元において、0でないイデアル N_s を法とする剰余類の完全代表系として、2次元ベクトルで張られる平行四辺形の内部と境界上の格子点とすることを要旨とする公開鍵暗号による認証文生成装置である。

【0053】また、請求項27記載の発明は、請求項22ないし請求項24のいずれか1項記載の認証文生成方法により生成された暗号化認証子データを復号化し、平文でありこの暗号化認証子データに相当する認証文データの正当性を検証する認証文検証方法であって、暗号化された認証子ブロックデータに対して、第2の公開鍵データ値である e_o または e_s をべき乗した値を計算し、この計算値に対して、第1の公開鍵であるイデアル N_o または N_s を法とした剰余を求めることによって、認証

子データを復号化し、この復号化した認証子データと、前記平文の認証子データとが一致していた場合は、認証ないし検証過程を正当と判定し、この復号化した認証子データと、前記平文の認証子データとが不一致であった場合は、認証ないし検証過程を正当でないと判定する、ことを要旨とする公開鍵暗号による認証文検証方法である。

【0054】また、請求項28記載の発明は、請求項27に記載の公開鍵暗号による認証文検証方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な m 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする。

【0055】また、請求項29記載の発明は、請求項27に記載の公開鍵暗号による認証文検証方法において、前記イデアルを法とする剰余類の取り方は、ユークリッド的な2次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、2次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする。

【0056】また、請求項30記載の発明は、暗号化された認証子ブロックデータに対して、第2の公開鍵データ値である e_0 をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵であるイデアル N_0 を法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、を備える認証文検証装置であって、前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、ユークリッド的な m 次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、 m に対するオイラー関数の値を次元とする多次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする公開鍵暗号による認証文検証装置である。

【0057】また、請求項31記載の発明は、暗号化された認証子ブロックデータに対して、第2の公開鍵データ値である e_0 をべき乗した値を計算するべき乗計算手段と、この計算値に対して、第1の公開鍵であるイデアル N_0 を法とした剰余を求めて復号化された認証子データを得る剰余類計算手段と、この復号化された認証子データおよびこれに対応する平文の認証子データの比較に基づいて認証判定を行う判定手段と、を備える認証文検証装置であって、前記剰余類計算手段における前記イデアルを法とする剰余類の取り方は、ユークリッド的な2次元分体において、0でないイデアル N_0 を法とする剰余類の完全代表系として、2次元ベクトルで張られる超平行四辺体の内部と境界上の格子点とすることを要旨とする。

公開鍵暗号による認証文検証装置である。

【0058】また、請求項32記載の発明は、請求項15、請求項20、請求項25、または請求項30のいずれか1項に記載の装置において、前記剰余類計算手段における剰余の計算は、前記 m に対するオイラー関数値に等しい数の各成分毎に行うことを要旨とする。

【0059】また、請求項33記載の発明は、請求項16、請求項21、請求項26、または請求項31のいずれか1項に記載の装置において、前記剰余類計算手段における剰余の計算は、2つの各成分毎に行うことを要旨とする。

【0060】【作用】本発明においては、代数体のイデアルの剰余類の構成法とべき乗の演算アルゴリズムを与えることにより、代数体上に拡張された公開鍵暗号方式を実現するための鍵生成方法、暗号化方法および復号化方法を具体的に構成し、また鍵生成装置、暗号化装置、および復号化装置を具体的に構成した。

【0061】すなわち、本発明に係る公開鍵暗号による鍵生成方法および鍵生成装置によれば、従来の有理整数環上の公開鍵暗号を円分体または2次元の整数環の upper 拡張した公開鍵暗号方式を提供することができる。

【0062】また、本発明に係る公開鍵暗号による暗号化方式、復号化方式、暗号化装置及び復号化装置によれば、円分体または2次元分体それぞれにおけるイデアルを法とするべき乗方法及びイデアルを法とするべき乗演算装置を提供することにより、RSA型の公開鍵暗号方式及び暗号化通信方式及び暗号化通信装置を提供することができる。

【0063】その結果、同報通信に対する安全性の向上したRSA型暗号を構成することができる。また、本発明による公開鍵暗号によれば、従来の楕円曲線上のRSA暗号に比べて暗号化速度を大幅に高速化することができる。

【0064】また、本発明に係る公開鍵暗号による認証文生成方法、認証文検証方法、認証文生成装置、及び認証文検証装置によれば、円分体または2次元分体それぞれにおけるイデアルを法とするべき乗方法及びイデアルを法とするべき乗演算装置を提供することにより、RSA型の公開鍵暗号方式による認証通信が行える。

【0065】また、本発明による公開鍵暗号方式を用いた暗号化通信装置は、認証にも適用でき、暗号化通信および認証通信の双方に对称使用が可能となる。

【0066】

【発明の実施の形態】

【本発明に係る公開鍵暗号方式の原理】次に、本発明に係る公開鍵暗号方式の原理を詳細に説明する。有理整数環において、素数 p と互いに素な任意の整数 a に対して、

【数8】

$$ap^{-1} \equiv 1 \pmod{p} \quad \dots (1.0)$$

式(1.0)で示されるフェルマーの小定理が成り立つことが知られている。

【0.0.6.7】任意の代数体の整数環 O に於いても、有理整数環 Z と同様にフェルマーの小定理が成り立つ。つまり、 O の素イデアルを P とすれば、 P と素な元 a に対して、

【数.9】

$$a^{NrmP-1} \equiv 1 \pmod{P} \quad \dots (1.1)$$

式(1.1)が成り立つ。ただし、 $NrmP$ はイデアル P のノルムとする。

【0.0.6.8】また、素イデアル P, Q に対して、

【数.1.0】

$$N = PQ \quad \dots (1.2)$$

$$L = LCM(NrmP-1, NrmQ-1) \quad \dots (1.3)$$

とする。ここで、フェルマーの小定理より、 $K \equiv 1 \pmod{L}$ をみたす K に対して、

【数.1.1】

$$a^K \equiv a \pmod{N} \quad \dots (1.4)$$

が成立する。この合同式(1.4)で、 $K \equiv e \cdot d \pmod{L}$ を満たす e, d を選び、 N の剰余類を第1の公開鍵、 e を第2の公開鍵、 d を第2の秘密鍵とすれば、

【数.1.2】

$$e \cdot d \equiv 1 \pmod{N} \quad \dots (1.5)$$

合同式(1.5)が成り立ち、公開鍵方式の暗号化および復号化が可能となる。本原理に基づき、本発明である代数体上に拡大したRSA型の公開鍵暗号が構成できる。

【0.0.6.9】【暗号化通信装置の実施形態】次に図面を参照して、本発明の実施の形態を詳細に説明する。図1は、本発明に係る公開鍵暗号による暗号化通信装置1.1の全体構成を説明するブロック図である。図1によれば、暗号化通信装置1.1は、暗号生成装置2.1、暗号化装置3.1、復号化装置4.1、および通信路5.1から構成されている。

【0.0.7.0】暗号生成装置2.1は、公開鍵1(n)、公開鍵2(e)、秘密鍵1(p)、(q)、および秘密鍵2(d)を生成する装置である。

【0.0.7.1】暗号化装置3.1は、平文 M を暗号化して暗号文(C_0, C_1, \dots, C_{r-1})を通信路5.1に送出する装置であり、平文 M を受け入れる平文入力部3.3、平文 M を一連の分割平文(M_0, M_1, \dots, M_{r-1})に分割する平文分割部3.5、および分割平文(M_0, M_1, \dots, M_{r-1})を暗号化して暗号文(C_0, C_1, \dots, C_{r-1})を得て、この暗号文を送信する暗号化処理部3.7を備えている。

【0.0.7.2】復号化装置4.1は、 r 個のブロックからなる暗号文(C_0, C_1, \dots, C_{r-1})を復号化して分割平文(M_0, M_1, \dots, M_{r-1})を得る復号化処理部4.3、この

$$(C_0, C_1, \dots, C_{r-1}) \equiv (M_0, M_1, \dots, M_{r-1}) e \pmod{N} \quad \dots (1.6)$$

式(1.6)によって暗号化を行なう。この暗号文は、通信路5.1を介して、受け手に送られる。

【0.0.8.0】【復号化装置】復号化装置4.1は、 r 個の

分割平文(M_0, M_1, \dots, M_{r-1})を統合して平文 M を得る平文統合処理部4.7、および平文 M を出力する平文出力部4.9を備えて構成されている。通信路5.1は、従来の通信路と同様の通信路である。

【0.0.7.3】次に、この暗号化通信装置1.1を構成する各装置の機能の概略を説明する。

【暗号生成装置】まず、暗号生成装置2.1は、素イデアル生成部2.3により2個の素イデアル P, Q (秘密鍵1)を生成し、その積 $N = PQ$ の剰余類(公開鍵1)を決定する。次いで、素イデアル P, Q から L を計算し、 e (公開鍵2)と、 d (秘密鍵2)を生成するものである。

【0.0.7.4】暗号生成装置2.1における暗号生成処理は、

【2個の素イデアル(秘密鍵1)の生成】、【剰余類(公開鍵1)の決定】及び【公開鍵2である e と秘密鍵2である d の生成】の3段階からなり、代数体が円分体であるか、2次体であるかによって、処理内容が異なる。

【0.0.7.5】【円分体における暗号生成処理】円分体の場合は、まず、円分体を生成する原始根の位数 m を入力として、互いに異なる2つの秘密鍵1である素イデアル P, Q を出力する。次いで、2つの秘密鍵1である素イデアル P, Q を入力として、公開鍵2である e と秘密鍵2である d を生成し出力する。

【0.0.7.6】【2次体における暗号生成処理】2次体の場合は、判別式 D を入力として、互いに異なる2つの秘密鍵1である素イデアル P, Q を出力する。次いで、2つの秘密鍵1である素イデアル P, Q を入力として、公開鍵2である e と秘密鍵2である d を生成し出力する。

【0.0.7.7】【暗号化装置】次に、暗号化装置3.1は、平文入力部3.3、平文分割処理部3.5、及び暗号化処理部3.7を含んで構成され、暗号化処理部3.7は、イデアル n を法とするべき乗演算部を含んでいる。

【0.0.7.8】平文入力部3.3は、送信すべきメッセージである平文 M を受け入れる。平文分割処理部3.5は、暗号生成装置2.1よりブロック数 r および分割平文長 $[1 \leq 2n]$ ($[x]$ は、 x を超えない最大の整数を示すガウス記号である)を得て、平文を一連の分割平文(M_0, M_1, \dots, M_{r-1})に分割する。ここで、 r をイデアル N の剰余類の完全代表系としてのベクトル空間 Z 上の整数環 O の次元とする。そして、平文 M に対して、それぞれのブロックが剰余類の範囲内の r 個のブロックからなる分割平文(M_0, M_1, \dots, M_{r-1})を生成する。

【0.0.7.9】暗号化処理部3.7は、イデアル n を法とするべき乗演算部3.9により、受け手の公開鍵 e を用いて、

【数.1.3】

ブロックからなる暗号文(C_0, C_1, \dots, C_{r-1})を復号化して分割平文(M_0, M_1, \dots, M_{r-1})を得る復号化処理部4.3、この分割平文(M_0, M_1, \dots, M_{r-1})を統合

して平文Mを得る平文統合処理部4.7、および平文Mを出力する平文出力部4.9を備えて構成されている。このうち復号化処理部4.3は、イデアル \mathfrak{n} を法とするべき乗演算部4.5を備えている。

$$(C_0, C_1, \dots, C_{r-1}), d := (M_0, M_1, \dots, M_{r-1}) \pmod{N} \quad (1.7)$$

式(1.7)を計算することによって、復号化を行なう。

【0082】平文統合処理部4.7は、復号化された分割平文 $(M_0, M_1, \dots, M_{r-1})$ を統合し、原平文Mへ復元し、平文出力部4.9から出力する。

【0083】次に、鍵生成装置2.1における本発明の公開鍵略号方式に用いられる公開鍵(略号化鍵)及び秘密鍵(復号化鍵)の生成方法を詳細に説明する。

【0084】【円分体における素イデアルの生成】ま

$$a_0 + a_1 \zeta_m + a_2 (\zeta_m)^2 + \dots + a_{\phi(m)-1} (\zeta_m)^{\phi(m)-1} \quad (1.8)$$

を $\mathbb{Z}[\phi(m)]$ の元 $(a_0, a_1, a_2, \dots, a_{\phi(m)-1})$ と同一視する。ただし、 $a_0, a_1, a_2, \dots, a_{\phi(m)-1} \in \mathbb{Z}$ であり、 $\phi(m)$ は1以上 $m-1$ 以下の自然数で m と互いに素な数の個数である(オイラー関数)。したが

$$p \nmid \phi(m) \pmod{m}, \quad p \not\equiv 1 \pmod{m}$$

であるとき \mathbb{Q} においても素数である。つまり、上の条件を満たす素数 p を選べば、その素数は円分体の整数環でも分解せず、素イデアル (p) の生成元 p となる。これを、惰性する素数と呼び、略号化装置の秘密鍵1とする。

【0086】具体例として、 $m=3, 5, 7, 11, 13$ の場合などは、素数 p として、

【数1.7】

$$m=3, \quad p \equiv 2 \pmod{3}$$

$$m=5, \quad p \equiv 2, 3 \pmod{5}$$

$$m=7, \quad p \equiv 3, 5 \pmod{7}$$

$$m=11, \quad p \equiv 2, 6, 7, 8 \pmod{11}$$

$$m=13, \quad p \equiv 2, 6, 7, 9, 11 \pmod{13}$$

を選べばよい。円分体は、原始 m 乗根の自然数 m が与えられると決定する。

【0087】図3は、円分体における素イデアル生成装置の機能図である。同図によれば、まず、円分体を生成する原始根の位数 m を入力し、次いで、有理素数 p を任意に生成する(ステップS1.0.1)。次いで、制御変数

$$\omega = (1 + \sqrt{-m}) / 2$$

$$\omega = \sqrt{-m}$$

とするとき、任意の整数 a, b を用いて、

【数1.9】

$$\alpha = a + b\omega \quad (2.2)$$

と書ける。そこで、2次体の整数環の元と2次元平面 \mathbb{Z}^2 を同一視する。したがって、2次体の整数環 \mathbb{O} は \mathbb{Z} 上2次元である。

【0090】また、2次体の判別式 D を m の4を法とする剰余の値に従って、

【数2.0】

$$D \equiv m, \quad m \equiv 1 \pmod{4} \quad (2.3)$$

【0081】復号化処理部4.3は、イデアル \mathfrak{n} を法とするべき乗演算部4.5により、受け手自身の秘密鍵 d を用いて、

【数1.4】

ず、円分体における素イデアルの生成について説明する。有理数体 \mathbb{Q} に1の原始 m 乗根 ζ_m (ただし、 m は素数のべきとする)を追加した体 $\mathbb{Q}(\zeta_m)$ を m 次円分体とよぶ。その整数環を \mathbb{O} と書く。 $m=3, 5, 7, 11, 13$ などのときは、ユークリッド環であった。また、この円分体の整数環 \mathbb{O} は $\mathbb{Z}[\phi(m)]$ に埋め込み可能であり、以降 \mathbb{O} の元。

【数1.5】

って、円分体の整数環 \mathbb{O} は \mathbb{Z} 上 $\phi(m)$ 次元である。

【0085】これより、円分体の整数環 \mathbb{O} において、有理素数 p は、

【数1.6】

$$(1 < \phi(m)) \dots (19)$$

1を1に初期設定する(ステップS1.0.3)。次いで、有理素数 p の1乗 $(\text{mod } m)$ を計算し、この値が1に等しいか否かを判定する(ステップS1.0.5)。

【0088】等しければ、ステップS1.0.1に戻り、再度新たな有理素数を生成する。等しくなければ、制御変数 i が m のオイラー関数値 $\phi(m)$ に等しいか否かを判定する(ステップS1.0.7)。等しければ、 p が素イデアル (p) の生成元 p となる条件を満たしているので、素イデアル (p) の探索を終了する。等しくなければ、制御変数 i を1だけ増加させて(ステップS1.0.9)、ステップS1.0.5に戻る。

【0089】【2次体における素イデアルの生成】次に、2次体における素イデアルの生成について説明する。有理数体 \mathbb{Q} に、二乗の因子を持たない有理整数 m の平方根を追加して得られる代数体 $\mathbb{Q}(\sqrt{m})$ を2次体と言う。2次体の整数環を \mathbb{O} とすると、 \mathbb{O} の全ての元 α は、

【数1.8】

$$m \equiv 1 \pmod{4} \quad (2.0)$$

$$m \equiv 2, 3 \pmod{4} \quad (2.1)$$

$$D = 4m, \quad m \equiv 2, 3 \pmod{4} \quad (2.4)$$

と定稿する。このとき、偶数でなく判別式で割れない素数 p に対して、

【数2.1】

$$(D/p)^2 = -1 \quad (2.5)$$

を満たす素数 p は、平方非剰余であり、2次体の整数環 \mathbb{O} の上でも分解せず素イデアル (p) の生成元となる。ここで、

【数2.2】 $(\mathbb{O}/p)^2$

は、平方剰余記号(ルジャンドルの記号とも呼ばれる)。

であり、例えばユークリッドの互除法を用いて計算できる。よって、上の性質を持つ素数を秘密鍵1として利用する。この素数も暗性する素数と呼ぶ。2次体は、判別式Dが与えられと決定することに注意する。

【0091】図4は、2次体における素イデアル生成装置の機能図である。同図によれば、まず、2次体を生成する判別式Dの値を入力し、次いで、有理素数pを任意に生成する(ステップS201)。次いで、平方剰余記号 $(D/p)_2$ を計算し、この値が-1であるか否かを判定する(ステップS203)。

【0092】 $(D/p)_2 = -1$ であれば、素数pは、2次体の整数環Oの上でも分解せず素イデアル(p)の生成元となるので、素イデアルpの探索を終了する。

$(D/p)_2 = -1$ でなければ、ステップS201へ戻り、 $L = \text{LCM}(p \phi(n) - 1, q \phi(n) - 1)$ を計算し、2次体の場合は

$$L = \text{LCM}(p^2 - 1, q^2 - 1)$$

を求める。次に、円分体2次体ともに

$$e d \equiv 1 \pmod{L}$$

を満たす、e、dを求める。秘密鍵2はdとし、公開鍵2はeとする。円分体における、これらの鍵を生成する装置の機能図を図5に示す。

【0096】【平文分割処理部】次に、図6を参照して、暗号化装置31における平文分割処理部35の分割機能の詳細を説明する。本発明に係る公開鍵暗号方式によれば、一回の暗号化処理の対象となる分割平文のブロック数rは、m次元円分体の場合 $\phi(m)$ 個であり、2次体の場合は2個である。このため、平文分割処理部35により、平文Mをそれぞれr個のブロックからなる平文ブロックに分割し、分割平文 $(M_0, M_1, \dots, M_{r-1})$ として、暗号化処理部37に供給する必要がある。

【0097】図6によれば、平文分割処理部35の入力として、平文入力部33より平文Mのビット列が与えられる。また、鍵生成装置21より、ブロック数r、及び、1ブロックの長さ(ビット数)である分割平文長 $[1 \leq 2 \leq n]$ が与えられる。分割平文長 $[1 \leq 2 \leq n]$ は、 $1 \leq 2 \leq n$ を超えない最大の整数とする(ここで、前記2個の暗性する素数の積 $n = pq$)。以上が平文分割処理部35の初期状態であり、例えばメモリ上に各数値およびデータを記憶して引き渡される。

【0098】次いで、制御変数iを0に設定し(ステップS401)、Mの先頭より $[1 \leq 2 \leq n]$ ビットの長さ $(C_0, C_1, \dots, C_{r-1}) \equiv (M_0, M_1, \dots, M_{r-1}) \cdot e \pmod{N}$

次に、 $(C_0, C_1, \dots, C_{r-1})$ を分割暗号文とし、受け手に送信する。ただし、rは整数環OのZ上の次元とする。以下で、剰余類の決定とイデアルを法とするべき乗演算処理について詳述する。

【0103】【イデアルを法とする剰余類の決定】本発明では、暗号化・復号化を一意的に行なうため、イデアルを法とする剰余類の取り方を以下のように定める。

り、異なる有理素数pを任意に生成する。

【0093】円分体・2次体ともに、以上の方法で生成された素イデアル(p)は以下生成元と同一視し、場合によりpと記述する。

【0094】【秘密鍵1および公開鍵1の生成】素イデアル(p)が探索された方法と同様の方法により、素イデアル(q)も探索されるが、素イデアル(p)と異なる素イデアル(q)を探索する必要がある。秘密鍵1は、これら2個の暗性する素数p、qとする。公開鍵1は、それらの積 $n = pq$ とする。

【0095】【秘密鍵2および公開鍵2の生成】秘密鍵1: p、qに対して、LCMを最小公倍数を求める関数とするとき、m次元円分体の場合は、

$$\begin{aligned} \text{【数23】} & \dots (26) \\ \text{【数24】} & \dots (27) \end{aligned}$$

$$\text{【数25】} \dots (28)$$

さのブロックを切り出し、この切り出されたブロックを M_i とし、残りのビット列を新たなMとする(ステップS403)。ブロック M_i は、例えばメモリ上に確保されたrの配列記憶位置のi番目に格納してもよい。

【0099】次いで、制御変数iが $r-1$ に等しいか否かを判定し(ステップS405)。この判定で $i \neq r-1$ であれば、iを1だけ増加させて(ステップS407)、残りのブロックの切り出しのために、ステップS403へ戻る。

【0100】ステップS405の判定において、 $i = r-1$ であれば、暗号化単位のrブロックの切り出しが終わったので、 M_0, M_1, \dots, M_{r-1} を暗号化処理部37へ出力する(ステップS409)。次いで、Mが空かどうかを判定し(ステップS411)。Mが空であれば、平文分割処理を終了し、Mが空でなければ、再度r個のブロックを切り出す為に、ステップS401へ戻る。以上のようにして、平文Mがブロック分割される。

【0101】【暗号化処理部】次に、暗号化処理部37の詳細を説明する。平文分割処理部35から入力された分割平文 $(M_0, M_1, \dots, M_{r-1})$ に対して、イデアルNを法とする以下のべき乗演算を行なう。

$$\begin{aligned} \text{【0102】} & \text{【数26】} \\ \text{【0104】} & \text{【円分体における剰余類】} \end{aligned}$$

まず、円分体におけるイデアルを法とする剰余類の取り方を説明する。ユークリッド的なm次元円分体においてOでないイデアルNによる剰余類の完全代表系として、 $Z[\phi(m)]$ に於いて原点からベクトル

$$\begin{aligned} \text{【数27】} & N, \zeta N, \zeta^2 N, \dots, \zeta^{\phi(m)-1} N \\ & \text{で張られる超平行四辺体の内部と境界上の格子点が取れ} \end{aligned}$$

る。ただし、境界上の格子点は原点に面さない超平面上の点は除く。

【0.1.0.5】また、任意の円分体において、惰性する素数の積で生成されるイデアル (n) による剰余類の完全代表系として $\mathbb{Z}[\phi(m)]$ に於いて、原点と

【数2.8】 $n, \tau n, \tau^2 n, \dots, \tau^{\phi(m)-1} n$ を頂点とする超立方体の内部と境界上の格子点が取れる。ただし、各軸を一边としない超平面上の点は除く。

【0.1.0.6】ここで、注意として惰性する素数の積で生成されるイデアル (n) による剰余類の計算方法は、各成分毎に有理整数の意味で $(\text{mod } n)$ を計算することである。

【0.1.0.7】【2次体における剰余類】次に、2次体におけるイデアルを法とする剰余類の取り方を説明する。ユークリッド的な2次体において0でないイデアル N による剰余類の完全代表系として、 \mathbb{Z}^2 に於いて、原点からベクトル

【数2.9】 $N, \omega N$

で張られる平行四辺形の内部と境界上の格子点が取れる。ただし、境界上の格子点は原点に面する超平面上の点のみ含める。

【0.1.0.8】また、任意の2次体において、惰性する素数の積で生成されるイデアル (n) による剰余類の完全代表系として、2次元平面 \mathbb{Z}^2 に於いて、原点と、

【数3.0】 $n, \omega n$

を頂点とする正方形の内部と、境界上の格子点、(各座標

$$x_{2n} = (x_n)^2, \quad x_{2n+1} = (x_n)^2 \cdot x, \quad \dots \quad (3.0)$$

式 (3.0) に示すように、乗算と2乗演算であるので、この乗算と2乗演算方法を以下に説明する。

【0.1.1.3】【円分体における乗算および2乗演算方法】まず、円分体における乗算および2乗演算方法について説明する。円分体に於ける2元 $x = (x_0, x_1, \dots, x_{\phi(m)-1})$ と $y = (y_0, y_1, \dots, y_{\phi(m)-1})$

$$x \cdot (x) \cdot y = (x \cdot (x) \cdot y_0, x \cdot (x) \cdot y_1, \dots, x \cdot (x) \cdot y_{\phi(m)-1}) \quad \dots (3.1)$$

$$x \cdot (x) \cdot x = (x \cdot (x) \cdot x_0, x \cdot (x) \cdot x_1, \dots, x \cdot (x) \cdot x_{\phi(m)-1}) \quad \dots (3.2)$$

ここで、イデアル (n) の剰余の取り方は、すべての成分の式を $(\text{mod } n)$ で計算する。これにより、イデアル (n) を法とするべき乗演算装置が構成できる。

【0.1.1.5】次に、円分体に於いて、拡大次数が低次でかつ素数の場合のアルゴリズムを詳細に記述する。特に、 $m=3$ の場合のべき乗演算を行なうための乗算と2乗演算装置を図7と図8に示す。図7及び図8において、符号3.0.1、3.0.3、3.1.3、3.1.5、3.2.5、3.

$$x \cdot (x) \cdot y = (x \cdot (x) \cdot y_0, x \cdot (x) \cdot y_1) \quad \dots (3.3)$$

ここで、

$$\begin{aligned} x \cdot (x) \cdot y_0 &= x_0 \cdot y_0 - x_1 \cdot y_1 \\ x \cdot (x) \cdot y_1 &= x_1 \cdot y_0 + (x_0 - x_1) \cdot y_1 \\ x \cdot (x) \cdot x &= (x \cdot (x) \cdot x_0, x \cdot (x) \cdot x_1) \quad \dots (3.4) \end{aligned}$$

が整数値となる点)が取れる。ただし、各軸を一边としない面上の点は除く。

【0.1.0.9】ここで、注意として惰性する素数の積で生成されるイデアル (n) による剰余類の計算方法は、各成分毎に有理整数の意味で $(\text{mod } n)$ を計算することである。

【0.1.1.0】また、任意の2次体において、惰性する素数の積で生成されるイデアル (n) による剰余類の完全代表系として、2次元平面 \mathbb{Z}^2 (各軸を次元1・次元2と呼ぶ) に於いて、原点 $(0, 0)$ と、 $(n, 0)$ 、 (n, n) 、 $(0, n)$ を頂点とする正方形の内部と、境界上の格子点 (各座標が整数値となる点) が取れる。ただし、各軸を一边としない面上の点は除く。この格子点を図3.2に示す。

【0.1.1.1】【イデアル (n) を法とするべき乗演算】次に、イデアル (n) を法とするべき乗演算について説明する。本発明の公開簡略号方式では、式 (1.6)、(1.7) に示したようなべき乗演算が行われる。べき乗演算では、指数部を2進表現して、この関数を指数部が2のべきとなる因数の積に分解し、小さい因数から順次、法による乗算を繰り返していく、反復平方根によるべき乗演算法、または2進計算法と呼ばれる既知の高速べき乗演算方法を使用することができる。

【0.1.1.2】このべき乗計算のための基本演算は、【数3.1】の乗算 $x \cdot (x) \cdot y$ 、および $x = (x_0, x_1, \dots, x_{\phi(m)-1})$ の2乗演算 $x \cdot (x) \cdot x$ は、それぞれ式 (3.1)、式 (3.2) に従って求めることができる。

【0.1.1.3】

【数3.2】

の乗算 $x \cdot (x) \cdot y$ 、および $x = (x_0, x_1, \dots, x_{\phi(m)-1})$ の2乗演算 $x \cdot (x) \cdot x$ は、それぞれ式 (3.1)、式 (3.2) に従って求めることができる。

【0.1.1.4】

【数3.2】

$$x \cdot (x) \cdot y = (x \cdot (x) \cdot y_0, x \cdot (x) \cdot y_1, \dots, x \cdot (x) \cdot y_{\phi(m)-1}) \quad \dots (3.1)$$

$$x \cdot (x) \cdot x = (x \cdot (x) \cdot x_0, x \cdot (x) \cdot x_1, \dots, x \cdot (x) \cdot x_{\phi(m)-1}) \quad \dots (3.2)$$

3.1及び3.3.3は、通常の乗算回路であり、符号3.0.5、3.1.1、3.2.3は減算回路、符号3.1.7及び3.2.1は加算回路、符号3.0.7、3.1.9、3.2.7、3.3.5は $\text{mod } n$ 演算回路である。

【0.1.1.6】【 $m=3$ の場合の乗算と2乗演算】 $m=3$ の場合、オイラー関数値 $\phi(m)$ は、 $\phi(3)=2$ となり、乗算および2乗演算は、

【数3.3】

$$x \cdot (x) \cdot y = (x \cdot (x) \cdot y_0, x \cdot (x) \cdot y_1) \quad \dots (3.3)$$

ここで、

$$\begin{aligned} x \cdot (x) \cdot y_0 &= x_0 \cdot y_0 - x_1 \cdot y_1 \\ x \cdot (x) \cdot y_1 &= x_1 \cdot y_0 + (x_0 - x_1) \cdot y_1 \\ x \cdot (x) \cdot x &= (x \cdot (x) \cdot x_0, x \cdot (x) \cdot x_1) \quad \dots (3.4) \end{aligned}$$

ここで、

$$x(x) \cdot x_0 = (x_0 + x_1)(x_0 - x_1)$$

$$x(x) \cdot x_1 = 2 \cdot x_0 \cdot x_1$$

それぞれ式(33)、(34)に示すとおりとなる。

$$x(x) \cdot y = (x(x) \cdot y_0, x(x) \cdot y_1, \dots, x(x) \cdot y_3) \dots (35)$$

ここで、

$$x(x) \cdot y_0 =$$

$$= x_3(y_2 - y_1) + x_2(y_3 - y_2) + x_0 y_0 - x_1 y_3$$

$$x(x) \cdot y_1$$

$$= x_1(y_0 - y_3) + (x_0 + x_3) y_1 - x_2 y_2 - x_3 y_3$$

$$x(x) \cdot y_2$$

$$= x_2(y_0 - y_2) + x_1(y_1 - y_3) + x_0 y_2 - x_3 y_1$$

$$x(x) \cdot y_3$$

$$= x_3(y_0 - y_1) + x_2(y_1 - y_2) + x_2 y_1 - x_1 y_3$$

$$x(x) \cdot x = (x(x) \cdot x_0, x(x) \cdot x_1, \dots, x(x) \cdot x_3) \dots (36)$$

ここで、

$$x(x) \cdot x_0 = (x_0 + x_2)(x_0 - x_2) + 2 \cdot x_3$$

$$(x_2 - x_1)$$

$$x(x) \cdot x_1 = 2 \cdot x_1(x_0 - x_3) + (x_3 + x_2)$$

$$(x_3 - x_2)$$

$$x(x) \cdot x_2 = x_1(x_1 - 2 \cdot x_3) + x_2(2 \cdot x_0 -$$

$$x_2)$$

$$x(x) \cdot y = (x(x) \cdot y_0, x(x) \cdot y_1, \dots, x(x) \cdot y_5) \dots (37)$$

ここで、

$$x(x) \cdot y_0 = x_0 y_0 - x_5 y_1 + (-x_4 + x_5)$$

$$y_2 + (-x_3 + x_4) y_3 + (-x_2 + x_3) y_4 +$$

$$(-x_1 + x_2) y_5$$

$$x(x) \cdot y_1 = x_1 y_0 - x_4 y_2 + (x_0 - x_5) y_1$$

$$+ (-x_3 + x_5) y_3 + (-x_2 + x_4) y_4 + (-$$

$$x_1 + x_3) y_5$$

$$x(x) \cdot y_2 = x_2 y_0 - x_3 y_3 + (x_1 - x_5) y_1$$

$$+ (x_0 - x_4) y_2 + (-x_2 + x_5) y_4 + (-$$

$$1 + 2 \cdot x_4) y_5$$

$$x(x) \cdot x = (x(x) \cdot x_0, x(x) \cdot x_1, \dots, x(x) \cdot x_5) \dots (38)$$

ここで、

$$x(x) \cdot x_0 = (x_0 + x_3)(x_1 - x_3) + 2 \cdot x_4$$

$$(x_3 + x_2) + 2 \cdot x_5(x_2 - x_1)$$

$$x(x) \cdot x_1 = 2 \cdot x_1(x_0 - x_5) + x_3(2 \cdot x_5 -$$

$$x_3) + x_4(x_4 - 2 \cdot x_2)$$

$$x(x) \cdot x_2 = (x_1 + x_3)(x_1 - x_3) + 2 \cdot x_5$$

$$(x_4 - x_1) + x_2(x_0 - x_4)$$

$$x(x) \cdot x_3 = x_5(x_5 - 2 \cdot x_1) + x_3(2 \cdot x_0 -$$

$$x_3) + 2 \cdot x_2(x_1 - x_4)$$

$$x(x) \cdot y = (x(x) \cdot y_0, x(x) \cdot y_1, \dots, x(x) \cdot y_9) \dots (39)$$

ここで、

$$x(x) \cdot y_0 = x_0 y_0 - x_9 y_1 + (-x_8 + y_9)$$

$$y_2 + (-x_7 + x_8) y_3 + (-x_6 + x_7) y_4 +$$

$$(-x_5 + x_6) y_5 + (-x_4 + x_5) y_6 + (-x_3$$

$$+ x_4) y_7 + (-x_2 + x_3) y_8 + (-x_1 + x_2$$

$$) y_9$$

【O.1.1.7】 [m=5の場合の乗算と2乗演算] m=5

の場合、オイラー関数値 $\phi(m)$ は、 $\phi(5)=4$ とな

り、乗算および2乗演算は、

【数3.4】

$$x(x) \cdot x_3 = x_2(2 \cdot x_1 - x_2) + 2 \cdot x_3(x_0 -$$

$$x_1)$$

それぞれ式(35)、(36)に示すとおりとなる。

【O.1.1.8】 [m=7の場合の乗算と2乗演算] m=7

の場合、オイラー関数値 $\phi(m)$ は、 $\phi(7)=6$ とな

り、乗算および2乗演算は、

【数3.5】

$$x(x) \cdot y_5 = x_5 y_0 + (x_4 - x_5) y_1 + (x_3$$

$$- x_4) y_2 + (x_2 - x_3) y_3 + (5 \cdot x_1 - x_2)$$

$$y_4 + (x_0 - x_1) y_5$$

$$x(x) \cdot y_4 = x_4 y_0 - 4 \cdot x_1 y_5 + (x_3 - x_5)$$

$$y_1 + (x_2 - x_4) y_2 + (x_1 - x_3) y_3 + (x_0$$

$$- x_2) y_4$$

$$x(x) \cdot y_3 = x_3 y_0 - 3 \cdot x_2 y_4 + (x_2 - x_5)$$

$$y_1 + (x_1 - x_4) y_2 + (x_0 - x_3) y_3 + (-x_1$$

$$+ x_5) y_5$$

$$x(x) \cdot y_2 = x_2 y_0 - 4 \cdot x_1 y_5 + (x_3 - x_5)$$

$$y_1 + (x_2 - x_4) y_2 + (x_1 - x_3) y_3 + (x_0$$

$$- x_2) y_4$$

$$x(x) \cdot y_1 = x_1 y_0 - x_4 y_2 + (x_0 - x_5) y_1$$

$$+ (-x_3 + x_5) y_3 + (-x_2 + x_4) y_4 + (-$$

$$x_1 + x_3) y_5$$

$$x(x) \cdot y_0 = x_0 y_0 - x_5 y_1 + (-x_4 + x_5)$$

$$y_2 + (-x_3 + x_4) y_3 + (-x_2 + x_3) y_4 +$$

$$(-x_1 + x_2) y_5$$

$$x(x) \cdot x = (x(x) \cdot x_0, x(x) \cdot x_1, \dots, x(x) \cdot x_5) \dots (38)$$

$$x(x) \cdot y = (x(x) \cdot y_0, x(x) \cdot y_1, \dots, x(x) \cdot y_9) \dots (39)$$

$$x(x) \cdot y_1 = x_1 y_0 - x_8 y_2 + (x_0 - x_9) y_1$$

$$+ (-x_7 + x_8) y_3 + (-x_6 + x_7) y_4 + (-$$

$$x_5 + x_6) y_5 + (-x_4 + x_5) y_6 + (-x_3 +$$

$$x_4) y_7 + (-x_2 + x_3) y_8 + (-x_1 + x_2)$$

$$y_9$$

$$x(x) \cdot y_2 = x_2 y_0 - x_7 y_3 + (x_1 - x_9) y_1$$

$$+ (-x_6 + x_7) y_4 + (-x_5 + x_6) y_5 + (-x_4 + x_5)$$

$$y_6 + (-x_3 + x_4) y_7 + (-x_2 + x_3) y_8 + (-x_1 + x_2)$$

$$y_9$$

$$x(x) \cdot y_3 = x_3 y_0 - 6 \cdot x_2 y_4 + (x_2 - x_5)$$

$$y_1 + (x_1 - x_4) y_2 + (x_0 - x_3) y_3 + (-x_1$$

$$+ x_5) y_5$$

$$x(x) \cdot y_4 = x_4 y_0 - 4 \cdot x_1 y_5 + (x_3 - x_5)$$

$$y_1 + (x_2 - x_4) y_2 + (x_1 - x_3) y_3 + (x_0$$

$$- x_2) y_4$$

$$x(x) \cdot y_5 = x_5 y_0 + (x_4 - x_5) y_1 + (x_3$$

$$- x_4) y_2 + (x_2 - x_3) y_3 + (5 \cdot x_1 - x_2)$$

$$y_4 + (x_0 - x_1) y_5$$

$$\begin{aligned}
& 1 + (x_0 - x_8) y_2 + (-x_6 + x_9) y_4 + (-x_5 + x_8) y_5 + (-x_4 + x_7) y_6 + (-x_3 + x_6) y_7 + (-x_2 + x_5) y_8 + (-x_1 + x_4) y_9 \\
x(x) y_3 &= x_3 y_0 - x_6 y_4 + (x_2 - x_9) y_1 + (-x_1 - x_8) y_2 + (x_0 - x_7) y_3 + (-x_5 + x_9) y_5 + (-x_4 + x_8) y_6 + (-x_3 + x_7) y_7 + (-x_2 + x_6) y_8 + (-x_1 + x_5) y_9 \\
x(x) y_4 &= x_4 y_0 - x_5 y_5 + (x_3 - x_9) y_1 + (x_2 - x_8) y_2 + (x_1 - x_7) y_3 + (x_0 - x_6) y_4 + (-x_4 + x_9) y_6 + (-x_3 + x_8) y_7 + (-x_2 + x_7) y_8 + (-x_1 + x_6) y_9 \\
x(x) y_5 &= x_5 y_0 - x_4 y_6 + (x_4 - x_9) y_1 + (x_3 - x_8) y_2 + (x_2 - x_7) y_3 + (x_1 - x_6) y_4 + (x_0 - x_5) y_5 + (-x_3 + x_9) y_7 + (-x_2 + x_8) y_8 + (-x_1 + x_7) y_9 \\
x(x) y_6 &= x_6 y_0 - x_3 y_7 + (x_5 - x_9) y_1 + x(x) x = (x(x), x_0, x(x), x_1, \dots, x(x), x_9) \dots (40)
\end{aligned}$$

ここで、

$$\begin{aligned}
x(x) x_0 &= (x_0 + x_5)(x_0 - x_5) + 2x_9(x_2 - x_1) + 2x_8(x_3 - x_2) + 2x_7(x_4 - x_3) + 2x_6(x_5 - x_4) \\
x(x) x_1 &= 2x_1(x_0 - x_9) + 2x_3(x_9 - x_7) + 2x_8(x_4 - x_2) + x_6(x_6 - 2x_4) + x_5(2x_7 - x_5) \\
x(x) x_2 &= 2x_2(x_0 - x_8) + 2x_4(x_9 - x_6) + x_1(x_1 - 2x_8) + x_5(2x_8 - x_5) + 2x_7(x_6 - x_3) \\
x(x) x_3 &= 2x_3(x_0 - x_7) + 2x_2(x_1 - x_8) + 2x_9(x_5 - x_1) + 2x_6(x_8 - x_4) + (x_9 + x_5)(x_9 - x_5) \\
x(x) x_4 &= 2x_4(x_0 - x_6) + 2x_3(x_1 - x_7) + 2x_9(x_6 - x_1) + 2x_8(x_7 - x_2) + (x_2 + x_5)(x_2 - x_5) \\
x(x) x_5 &= 2x_5(x_0 - x_5) + 2x_1(x_4 - x_6) + 2x_2(x_3 - x_8) + 2x_9(x_7 - x_1) \\
x(x) y &= (x(x), y_0, x(x), y_1, \dots, x(x), y_{11}) \dots (41)
\end{aligned}$$

ここで、

$$\begin{aligned}
x(x) y_0 &= x_0 y_0 - x_{11} y_1 + (-x_2 + x_3) y_{10} + (-x_1 + x_2) y_{11} + (-x_{10} + x_{11}) y_2 + (x_{10} - x_9) y_3 + (-x_8 + x_9) y_4 + (-x_7 + x_8) y_5 + (-x_6 + x_7) y_6 + (-x_5 + x_6) y_7 + (-x_4 + x_5) y_8 + (-x_3 + x_4) y_9 \\
x(x) y_1 &= x_1 y_0 - x_{10} y_2 + (x_0 - x_{11}) y_1 + (-x_2 + x_4) y_{10} + (-x_1 + x_3) y_{11} + (x_{11} - y_9) y_3 + (x_{10} - x_8) y_4 + (-x_7 + x_9) y_5 + (-x_6 + x_8) y_6 + (-x_5 + x_7) y_7 + (-x_4 + x_6) y_8 + (-x_3 + x_5) y_9 \\
x(x) y_2 &= x_2 y_0 - x_9 y_3 + (x_1 - x_{11}) y_1 + (-x_2 + x_5) y_{10} + (-x_1 + x_4) y_{11} + (x_0 - x_{10}) y_2 + (x_{11} - x_8) y_4 + (x_{10} - x_7)
\end{aligned}$$

$$\begin{aligned}
& 1 + (x_4 - x_8) y_2 + (x_3 - x_7) y_3 + (x_2 - x_6) y_4 + (x_1 - x_5) y_5 + (x_0 - x_4) y_6 + (-x_2 + x_9) y_8 + (-x_1 + x_8) y_9 \\
x(x) y_7 &= x_7 y_0 - x_2 y_8 + (x_6 - x_9) y_1 + (x_5 - x_8) y_2 + (x_4 - x_7) y_3 + (x_3 - x_6) y_4 + (x_2 - x_5) y_5 + (x_1 - x_4) y_6 + (x_0 - x_3) y_7 + (-x_1 + x_9) y_9 \\
x(x) y_8 &= x_8 y_0 - x_1 y_9 + (x_7 - x_9) y_1 + (x_6 - x_8) y_2 + (x_5 - x_7) y_3 + (x_4 - x_6) y_4 + (x_3 - x_5) y_5 + (x_2 - x_4) y_6 + (x_1 - x_3) y_7 + (x_0 - x_2) y_8 \\
x(x) y_9 &= x_9 y_0 + (x_8 - x_9) y_1 + (x_7 - x_8) y_2 + (x_6 - x_7) y_3 + (x_5 - x_6) y_4 + (x_4 - x_5) y_5 + (x_3 - x_4) y_6 + (x_2 - x_3) y_7 + (x_1 - x_2) y_8 + (x_0 - x_1) y_9
\end{aligned}$$

【数3.8】

$$\begin{aligned}
& + 2x_8(x_5 - x_2) \\
x(x) x_6 &= 2x_6(x_0 - x_4) + 2x_1(x_5 - x_4) + 2x_2(x_4 - x_8) + x_3(x_3 - 2x_7) + x_5(2x_1 - x_5) \\
x(x) x_7 &= 2x_7(x_0 - x_3) + 2x_1(x_6 - x_8) + x_5(2x_2 - x_5) + 2x_3(x_3 - x_7) + 2x_9(x_6 - x_1) \\
x(x) x_8 &= 2x_8(x_0 - x_3) + 2x_1(x_7 - x_9) + 2x_2(x_6 - x_8) + 2x_3(x_5 - x_3) + 2x_4(x_4 - x_6) \\
x(x) x_9 &= 2x_9(x_0 - x_1) + 2x_8(x_1 - x_2) + 2x_7(x_2 - x_3) + 2x_6(x_3 - x_4) + 2x_5(x_4 - x_5)
\end{aligned}$$

それぞれ式(39)、(40)に示すとおりとなる。

【0120】 $[m=13]$ の場合の乗算と2乗演算 $m=13$ の場合、オイラー関数値 $\phi(m)$ は、 $\phi(13)=12$ となり、乗算および2乗演算は、

【数3.9】

$$\begin{aligned}
& x(x) y = (x(x), y_0, x(x), y_1, \dots, x(x), y_{11}) \dots (41) \\
& y_5 + (-x_6 + x_9) y_6 + (-x_5 + x_8) y_7 + (-x_4 + x_7) y_8 + (-x_3 + x_6) y_9 \\
x(x) y_3 &= x_3 y_0 - x_8 y_4 + (-x_{11} + x_2) y_1 + (-x_2 + x_6) y_{10} + (-x_1 + x_5) y_{11} + (x_1 - x_{10}) y_2 + (x_0 - x_9) y_3 + (x_{11} - x_7) y_5 + (x_{10} - x_6) y_6 + (-x_5 + x_9) y_7 + (-x_4 + x_8) y_8 + (-x_3 + x_7) y_9 \\
x(x) y_4 &= x_4 y_0 - x_7 y_5 + (-x_{11} + x_3) y_1 + (-x_2 + x_7) y_{10} + (-x_1 + x_6) y_{11} + (-x_{10} + x_2) y_2 + (x_1 - x_9) y_3 + (x_0 - x_8) y_4 + (x_{11} - x_6) y_6 + (x_{10} - x_5) y_7 + (-x_4 + x_9) y_8 + (-x_3 + x_8) y_9 \\
x(x) y_5 &= x_5 y_0 - x_6 y_6 + (-x_{11} + x_4) y_1 + (-x_2 + x_8) y_{10} + (-x_1 + x_7) y_{11} +
\end{aligned}$$

$$\begin{aligned}
& (-x_{10}+x_3) \cdot y_2 + (x_2-x_9) \cdot y_3 + (x_1-x_8) \cdot y_4 + (x_0-x_7) \cdot y_5 + (x_{11}-x_5) \cdot y_7 \\
& + (x_{10}-x_4) \cdot y_8 + (-x_3+x_9) \cdot y_9 \\
x \cdot (x) \cdot y_6 = & x_6 \cdot y_0 - x_5 \cdot y_7 + (-x_{11}+x_5) \cdot y_1 + (-x_2+x_9) \cdot y_{10} + (-x_1+x_8) \cdot y_{11} + \\
& (-x_{10}+x_4) \cdot y_2 + (x_3-x_9) \cdot y_3 + (x_2-x_8) \cdot y_4 + (x_1-x_7) \cdot y_5 + (x_0-x_6) \cdot y_6 \\
& + (x_{11}-x_4) \cdot y_8 + (x_{11}-x_4) \cdot y_8 + (x_{10}-x_3) \cdot y_9 \\
x \cdot (x) \cdot y_7 = & x_7 \cdot y_0 - x_4 \cdot y_8 + (-x_{11}+x_6) \cdot y_1 + (x_{10}-x_2) \cdot y_{10} + (-x_1+x_9) \cdot y_{11} + \\
& (-x_{10}+x_5) \cdot y_2 + (x_4-x_9) \cdot y_3 + (x_3-x_8) \cdot y_4 + (x_2-x_7) \cdot y_5 + (x_1-x_6) \cdot y_6 \\
& + (x_0-x_5) \cdot y_7 + (x_{11}-x_3) \cdot y_9 \\
x \cdot (x) \cdot y_8 = & x_8 \cdot y_0 - x_3 \cdot y_9 + (-x_{11}+x_7) \cdot y_1 + (x_{11}-x_2) \cdot y_{10} + (-x_1+x_{10}) \cdot y_{11} + \\
& (-x_{10}+x_6) \cdot y_2 + (x_5-x_9) \cdot y_3 + (x_4-x_8) \cdot y_4 + (x_3-x_7) \cdot y_5 + (x_2-x_6) \cdot y_6 \\
& + (x_1-x_5) \cdot y_7 + (x_0-x_4) \cdot y_8 + (x_{11}-x_3) \cdot y_9 \\
x \cdot (x) \cdot x = & (x \cdot (x) \cdot x_0, x \cdot (x) \cdot x_1, \dots, x \cdot (x) \cdot x_{11}) \dots (42)
\end{aligned}$$

ここで、

$$\begin{aligned}
x \cdot (x) \cdot x_0 = & (x_0+x_6) \cdot (x_0-x_6) + 2 \cdot x_{11} \cdot (x_2-x_{11}) + 2 \cdot x_{10} \cdot (x_3-x_2) + 2 \cdot x_9 \cdot (x_4-x_3) + 2 \cdot x_5 \cdot (x_8-x_7) + 2 \cdot x_7 \cdot (x_6-x_5) \\
x \cdot (x) \cdot x_1 = & 2 \cdot x_1 \cdot (x_0-x_{11}) + 2 \cdot x_3 \cdot (x_{11}-x_9) + 2 \cdot x_4 \cdot (x_{10}-x_8) + 2 \cdot x_5 \cdot (x_9-x_3) + x_6 \cdot (2 \cdot x_8-x_6) + x_7 \cdot (x_7-2 \cdot x_5) \\
x \cdot (x) \cdot x_2 = & 2 \cdot x_2 \cdot (x_0-x_{10}) + x_1 \cdot (x_1-x_{11}) + 2 \cdot x_4 \cdot (x_{11}-x_8) + 2 \cdot x_5 \cdot (x_{10}-x_7) + 2 \cdot x_6 \cdot (x_9-x_6) + 2 \cdot x_7 \cdot (x_8-x_5) \\
x \cdot (x) \cdot x_3 = & 2 \cdot x_3 \cdot (x_0-x_9) + 2 \cdot x_{11} \cdot (x_5-x_1) + 2 \cdot x_2 \cdot (x_1-x_{10}) + x_6 \cdot (2 \cdot x_{10}-x_6) + 2 \cdot x_7 \cdot (x_9-x_5) + x_8 \cdot (x_8-2 \cdot x_4) \\
x \cdot (x) \cdot x_4 = & (x_2+x_6) \cdot (x_2-x_6) + 2 \cdot x_7 \cdot (x_{10}-x_5) + 2 \cdot x_{11} \cdot (x_6-x_1) + x_2 \cdot (x_2-2 \cdot x_{10}) + 2 \cdot x_3 \cdot (x_1-x_9) + 2 \cdot x_4 \cdot (x_0-x_8) \\
x \cdot (x) \cdot x_5 = & 2 \cdot x_5 \cdot (x_0-x_7) + 2 \cdot x_1 \cdot (x_4-x_{11}) + 2 \cdot x_2 \cdot (x_3-x_4) + 2 \cdot x_7 \cdot (x_{11}-x_5) + 2 \cdot x_{10} \cdot (x_8-x_2) + x_9 \cdot (x_9-2 \cdot x_3) \\
x \cdot (x) \cdot x_6 = & x_6 \cdot (2 \cdot x_0-x_6) + 2 \cdot x_1 \cdot (x_5-x_{11}) + 2 \cdot x_2 \cdot (x_4-x_{10}) + x_3 \cdot (x_3-2 \cdot x_9) + 2 \cdot x_8 \cdot (x_{11}-x_4) + 2 \cdot x_9 \cdot (x_{10}-x_3) \\
x \cdot (x) \cdot x_7 = & 2 \cdot x_7 \cdot (x_0-x_5) + 2 \cdot x_1 \cdot (x_6-x_{11}) + 2 \cdot x_2 \cdot (x_5-x_{10}) + 2 \cdot x_3 \cdot (x_4-x_9)
\end{aligned}$$

式 (4.3) となる。

$$\begin{aligned}
& (x_1-x_5) \cdot y_7 + (x_0-x_4) \cdot y_8 \\
x \cdot (x) \cdot y_9 = & x_9 \cdot y_0 - x_2 \cdot y_{10} + (-x_{11}+x_8) \cdot y_1 + (-x_1+x_{11}) \cdot y_{11} + (-x_{10}+x_7) \cdot y_2 + \\
& (x_6-x_9) \cdot y_3 + (x_5-x_8) \cdot y_4 + (x_4-x_7) \cdot y_5 + (x_3-x_6) \cdot y_6 + (x_2-x_5) \cdot y_7 + \\
& (x_1-x_4) \cdot y_8 + (x_0-x_3) \cdot y_9 \\
x \cdot (x) \cdot y_{10} = & x_{10} \cdot y_0 - x_1 \cdot y_{11} + (-x_{11}+x_9) \cdot y_1 + (x_0-x_2) \cdot y_{10} + (-x_{10}+x_8) \cdot y_2 + (x_7-x_9) \cdot y_3 + \\
& (x_6-x_8) \cdot y_4 + (x_5-x_7) \cdot y_5 + (x_4-x_6) \cdot y_6 + (x_3-x_5) \cdot y_7 + (x_2-x_4) \cdot y_8 + (x_1-x_3) \cdot y_9 \\
x \cdot (x) \cdot y_{11} = & x_{11} \cdot y_0 + (x_{10}-x_{11}) \cdot y_1 + (x_1-x_2) \cdot y_{10} + (x_0-x_1) \cdot y_{11} + (-x_{10}+x_9) \cdot y_2 + \\
& (x_8-x_9) \cdot y_3 + (x_7-x_8) \cdot y_4 + (x_6-x_7) \cdot y_5 + (x_5-x_6) \cdot y_6 + (x_4-x_5) \cdot y_7 + \\
& (x_3-x_4) \cdot y_8 + (x_2-x_3) \cdot y_9
\end{aligned}$$

【数 4.0】

$$\begin{aligned}
& + 2 \cdot x_8 \cdot (x_{11}-x_3) + x_{10} \cdot (x_{10}-2 \cdot x_2) \\
x \cdot (x) \cdot x_8 = & 2 \cdot x_8 \cdot (x_0-x_4) + 2 \cdot x_3 \cdot (x_7-x_{11}) + 2 \cdot x_2 \cdot (x_6-x_{10}) + 2 \cdot x_3 \cdot (x_5-x_9) + x_4 \cdot (x_4-2 \cdot x_8) + 2 \cdot x_{10} \cdot (x_{11}-x_2) \\
x \cdot (x) \cdot x_9 = & 2 \cdot x_9 \cdot (x_0-x_3) + 2 \cdot x_1 \cdot (x_8-x_{11}) + 2 \cdot x_2 \cdot (x_7-x_{10}) + 2 \cdot x_3 \cdot (x_6-x_9) + 2 \cdot x_4 \cdot (x_5-x_8) + x_{11} \cdot (x_{11}-2 \cdot x_1) \\
x \cdot (x) \cdot x_{10} = & 2 \cdot x_{10} \cdot (x_0-x_2) + 2 \cdot x_1 \cdot (x_9-x_{11}) + 2 \cdot x_2 \cdot (x_8-x_{10}) + 2 \cdot x_3 \cdot (x_7-x_9) + 2 \cdot x_4 \cdot (x_6-x_8) + x_5 \cdot (x_5-2 \cdot x_7) \\
x \cdot (x) \cdot x_{11} = & 2 \cdot x_{11} \cdot (x_0-x_1) + 2 \cdot x_{10} \cdot (x_1-x_2) + 2 \cdot x_9 \cdot (x_2-x_3) + 2 \cdot x_3 \cdot (x_8-x_9) + 2 \cdot x_7 \cdot (x_4-x_5) + x_6 \cdot (2 \cdot x_5-x_6)
\end{aligned}$$

それぞれ式 (4.1)、(4.2) に示すとおりとなる。

【0.1.2.1】 2 次体における乗算および 2 乗演算方法 次に、2 次体におけるイデアル (n) の剰余の取り方を説明する。判別式が 0 である、2 次体の任意の 2 元 $x = (x_0, x_1)$, $y = (y_0, y_1)$ に対する乗算 $x \cdot (x) \cdot y = (x \cdot (x) \cdot y_0, x \cdot (x) \cdot y_1)$ と、任意の元の $x = (x_0, x_1)$ の 2 乗演算 $x \cdot (x) \cdot x = (x \cdot (x) \cdot x_0, x \cdot (x) \cdot x_1)$ を行なう方法を記述する。ここで、イデアル (n) の剰余の取り方は、すべての成分の式を (mod: n) で計算する。これにより、イデアル (n) を法とするべき乗演算装置が構成できる。

【0.1.2.2】 まず、 $m \equiv 1 \pmod{4}$ の場合、

【数 4.1】

$$\begin{aligned}
x \cdot (x) \cdot y_0 = & x_0 \cdot y_0 + (m-1) \cdot (1/4) \cdot x_1 \cdot y_1 \\
x \cdot (x) \cdot y_1 = & x_1 \cdot (y_0 + (m-1) \cdot (1/4) \cdot y_1) + x_0 \cdot y_1 \\
x \cdot (x) \cdot x_0 = & (x_0)^2 + (m-1) \cdot (1/4) \cdot (x_1)^2 \\
x \cdot (x) \cdot x_1 = & x_1 \cdot (x_0 + (m-1) \cdot (1/4) \cdot x_1) \dots (43)
\end{aligned}$$

【0.1.2.3】 次に、 $m \equiv 2, 3 \pmod{4}$ の場合、

【数4.2】

$$\begin{aligned}x(x) \cdot y_0 &= x_0 \cdot y_0 + m \cdot x_1 \cdot y_1 \\x(x) \cdot y_1 &= x_1 \cdot (y_0 + m \cdot x_1) + x_0 \cdot y_1 \\x(x) \cdot x_0 &= (x_0) \cdot 2 + m \cdot (x_1) \cdot 2 \\x(x) \cdot x_1 &= x_1 \cdot (x_0 + m \cdot x_1) \quad \dots (44)\end{aligned}$$

式(4.4)となる。

【0.1.2.4】【復号化処理部】次に、復号化装置4.1の復号化処理部4.3の動作を説明する。復号化処理部4.3

$$(C_0, C_1, \dots, C_{r-1}) \cdot d \equiv (M_0, M_1, \dots, M_{r-1}) \pmod{(n)} \quad \dots (45)$$

式(4.5)を計算することによって、分割平文(M0, M1, ..., Mr-1)を復元する。

【0.1.2.5】復号化処理部4.3のイデアル(n)を法とするべき乗演算部4.5は、秘密鍵2.eによりべき乗演算を行うこと以外に、暗号化装置3.1のイデアル(n)を法とするべき乗演算部3.7と同じであり、暗号化装置3.1と復号化装置4.1の大部分は共通であり、暗号化・復号化装置として構成することにより、双方向の暗号化通信を行うことができる。

【0.1.2.6】【平文統合処理部】平文統合処理部4.7は、復号処理部によって得られる分割平文(M0, M1, ..., Mr-1)を順番に連結し原平文Mを得る。

【0.1.2.7】次に、フローチャート図およびメモリ上のデータ配置を示す表を参照しながら、本発明の実施形態を詳細に説明する。図9ないし図1.2は、円分体暗号における鍵生成の詳細手順を示すフローチャート図である。

【0.1.2.8】円分体暗号における鍵生成では、表1に示すようなメモリ上のデータ配置を行う。

【0.1.2.9】

【表1】

データ番号	データ内容
1	円分体の既約根の位数m
2	オイラー関数値φ(m)
3	有素数p
4	素イデアル1(秘密鍵1) p
5	素イデアル2(秘密鍵2) q
6	公開鍵 n = p・q
7	最小公倍数 L
8	暗号鍵 d
9	公開鍵 e

そして、図9に示すように、まず、円分体を生成する原始根の位数mをデータ番号1に入力する(ステップS5.0.1)。次いで、オイラー関数値φ(m)を計算し(ステップS5.1.0)、オイラー関数値φ(m)をデータ番号2に格納する(ステップS5.3.1)。次いで、素イデアル1(秘密鍵1) pを生成し(ステップS5.4.0)、素イデアル2(秘密鍵2) qを生成し(ステップS5.5.1)。

は、通信路5.1を介して暗号化装置3.1より送られてきた分割暗号文(C0, C1, ..., Cr-1)に対して、

【数4.3】

$$\dots (45)$$

0)、素イデアルp, qからその積である公開鍵1, n = p・qを求めて、データ番号6に格納する(ステップS5.3.1)。

【0.1.3.0】次いで、pφ(m)-1とqφ(m)-1との最小公倍数であるLを計算し、これをデータ番号7に格納する(ステップS5.3.3)。次いで、e・d≡1(mod L)となる2数、eおよびdを求め、秘密鍵2.dをデータ番号8に、公開鍵2.eをデータ番号9に、それぞれ格納して(ステップS5.3.5)、鍵生成を終了する。

【0.1.3.1】図1.0は、オイラー関数値φ(m)の計算ルーチンを示すフローチャートであり、ステップS5.1.0以下の処理の詳細を示す。まず、mを入力し(ステップS5.1.2)、次いで制御変数i, jをそれぞれ、1, 0に初期設定する(ステップS5.1.4)。次いで、iとmとの公約数が存在するか否かを判定する(ステップS5.1.6)。ステップS5.1.6の判定において、y.e.sであれば(公約数が存在すれば)なにもせず、n.oであれば(公約数が存在しなければ)jを1だけ増加させて(ステップS5.1.8)、ステップS5.2.0へ移る。

【0.1.3.2】次いで、iを1だけ増加させて(ステップS5.2.0)、iとmとが等しいか否かを判定する(ステップS5.2.2)。ステップS5.2.2の判定において、iとmとが等しくなければ、ステップS5.1.6へ戻り、iとmとが等しいければ、φ(m)=jとして計算を終了し(ステップS5.2.4)、次の処理へ移る。

【0.1.3.3】図1.1は、素イデアル1, pの生成処理を示すフローチャートであり、ステップS5.4.0以下の処理の詳細を示す。まず、所定の範囲の有理整数pを任意に(ランダムに)生成し、データ番号3に格納する(ステップS5.4.2)。次いで、制御変数iを1に初期設定する(ステップS5.4.4)。次いで、pi(mod m)を計算し、この値が1であるか否かを判定する(ステップS5.4.6)。

【0.1.3.4】ステップS5.4.6の判定で、pi≡1(mod m)であれば、ステップS5.4.2に戻り、再度有理素数pを生成する。pi≡1(mod m)でなければ、i=φ(m)か否かを判定する(ステップS5.4.8)。ステップS5.4.8の判定において、i=φ(m)でなければ、iに1を加えて(ステップS5.5.0)、ステップS5.4.6へ戻る。i=φ(m)であれば、pは素イデアルであると判定できるので、素イデアル1(秘密

【0135】図12は、素イデアル2、 q の生成処理を示すフローチャートであり、ステップS560以下の処理の詳細を示す。まず、所定の範囲の有理整数 a を任意に(ランダムに)生成する(ステップS562)。次いで、 a が p に等しいか否かを判定する(ステップS564)。 a と p とが等しければ、 p と異なる有理整数 a を発生させるために、ステップS562に戻る。 a が p と異なれば、次いで、制御変数 i を1に初期設定する(ステップS566)。次いで、 $qi \pmod{m}$ を計算し、この値が1であるか否かを判定する(ステップS568)。

【0136】ステップS568の判定で、 $qi \pmod{m}$ であれば、ステップS562に戻り、再度有理整数 a を生成する。 $qi \pmod{m}$ でなければ、 $i = \phi(m)$ か否かを判定する(ステップS570)。ステップS570の判定において、 $i = \phi(m)$ でなければ、 i に1を加えて(ステップS572)、ステップS568に戻る。 $i = \phi(m)$ であれば、 a は素イデアルであると判定できるので、素イデアル2(秘密鍵1) q をデータ番号5に格納して(ステップS574)、次の処理へ移る。

【0137】図13ないし図16は、2次体略号における鍵生成の詳細手順を示すフローチャートである。

【0138】2次体略号における鍵生成では、表2に示すようなメモリ上のデータ配置を行う。

【0139】

【表2】

データ番号	データの配置
1	2乗因子を持たない有理整数 m
2	判別式 D
3	有理素数 p
4	素イデアル1(秘密鍵1) p
5	素イデアル2(秘密鍵1) q
6	公開鍵1 $n = p \cdot q$
7	最小公倍数 l
8	秘密鍵2 d
9	公開鍵2 e

そして、図13に示すように、まず2乗因子を持たない有理整数 m をデータ番号1に入力する(ステップS601)。次いで、判別式 D の値をデータ番号2に格納する(ステップS610)。次いで、素イデアル1(秘密鍵1) p を生成して、データ番号4に格納し(ステップS620)、次いで、素イデアル2(秘密鍵1) q を生成して、データ番号5に格納する(ステップS640)。次いで、素イデアル p 、 q からその積である公開鍵1、

$n = p \cdot q$ を求めて、データ番号6に格納する(ステップS661)。

【0140】次いで、 $p2 = 1$ と $q2 = 1$ との最小公倍数である L を計算し、これをデータ番号7に格納する(ステップS663)。次いで、 $e \cdot d = 1 \pmod{L}$ となる2数 e および d を求め、秘密鍵2 d をデータ番号8に、公開鍵2 e をデータ番号9に、それぞれ格納して(ステップS665)、鍵生成を終了する。

【0141】図14は、判別式 D の値を格納するルーチンの詳細を示すフローチャートであり、ステップS610以下の詳細を示す。まず、データ番号1から有理整数 m を読み出し、 $m \equiv 1 \pmod{4}$ か否かを判定する(ステップS612)。 $m \equiv 1 \pmod{4}$ であれば、判別式 $D = m$ とし(ステップS6142)、 $m \equiv 1 \pmod{4}$ でなければ($m \equiv 2, 3 \pmod{4}$)、判別式 $D = 4m$ として(ステップS616)、判別式 D の値をデータ番号2に格納し(ステップS618)、次の処理へ移る。

【0142】図15は、2次体における素イデアル1、 p の生成処理の詳細を示すフローチャートであり、ステップS620以下の詳細を示す。まず、所定の範囲の有理整数 p を任意に(ランダムに)生成し、データ番号3に格納する(ステップS622)。次いで、ルジャンドルの記号 $(D/p)_2$ を計算する(ステップS624)。 $(D/p)_2$ の値は、例えばユークリッドの互除法を用いて計算することができる。

【0143】次いで、この値が-1であるか否かを判定する(ステップS626)。この判定で、 $(D/p)_2 = -1$ でなければ、ステップS622に戻り、再度有理素数 p を生成する。ステップS626の判定で、 $(D/p)_2 = -1$ であれば、 p は素イデアルであると判定できるので、素イデアル1(秘密鍵1) p をデータ番号4に格納して(ステップS628)、次の処理へ移る。

【0144】図16は、2次体における素イデアル2、 q の生成処理の詳細を示すフローチャートであり、ステップS640以下の詳細を示す。まず、所定の範囲の有理整数 a を任意に(ランダムに)生成する(ステップS642)。次いで、 a が p に等しいか否かを判定し(ステップS644)、等しければ p と異なる a を生成するために、ステップS642に戻る。等しくなければ、次いで、ルジャンドルの記号 $(D/p)_2$ を計算し(ステップS646)、この値が-1であるか否かを判定する(ステップS648)。

【0145】ステップS648の判定で、 $(D/p)_2 = -1$ でなければ、ステップS642に戻り、再度有理素数 a を生成する。ステップS648の判定で、 $(D/p)_2 = -1$ であれば、 a は素イデアルであると判定できるので、素イデアル2(秘密鍵1) q をデータ番号5に格納して(ステップS650)、次の処理へ移る。

【0146】図17ないし図20は、暗号化送信処理の

詳細手順を示すフローチャートである。暗号化送信処理では、表3に示すようなメモリ上のデータ配置を行う。

【0147】

【表3】

データ番号	データ内容
1	公開鍵 n
2	公開鍵 e
3	ブロック数 r
4	分割平文長 $(1 \leq i \leq 2n)$
5	平文 M
6	暗号化ブロック M_i
...	...
$5+r$	暗号化ブロック M_{i+1}
$6+r$	暗号化ブロック C_i
...	...
$5+2r$	暗号化ブロック C_{i+1}

そして、図17に示すように、まずステップS700で、暗号化のための初期設定が行われる。すなわちステップS701では、公開鍵 n 、公開鍵 e 、ブロック数 r 、分割平文長 $(1 \leq i \leq 2n)$ をそれぞれデータ番号1～4に設定する。

【0148】次いで、平文 M をデータ番号5に読み込む（ステップS703）。次いで、平文 M の先頭からそれぞれ分割平文長のビット数の長さのブロックを r 個切り出す平文分割処理を行い（ステップS710）、この r 個のブロック毎に暗号化を行う（ステップS730）。暗号化されたブロックは、暗号送信処理され（ステップS750）。また平文が残っていれば、平文分割処理（ステップS710）に戻り、平文が残っていなければ終了する。

【0149】図18は、平文分割処理の詳細を示すフローチャートであり、ステップS710以下の詳細を示す。まず、平文分割処理の制御変数 i 、 j 、 k の初期設定を行う（ステップS712）。制御変数 i は、分割された平文ブロックの番号を示すものである。制御変数 j は、元の平文データ M を構成する各ビットのビット番号である。 k は、ビット数で表した分割平文長である。ステップS712では、それぞれ、 $i=0$ 、 $j=0$ 、 $k=(1 \leq i \leq 2n)$ と設定する。

【0150】次いで、データ番号5から M を読み出し、 M の j ビット目から $(j+k-1)$ ビット目までを切り出し、これを M_i としてデータ番号 $(6+i)$ に格納する（ステップS714）。次いで、制御変数更新のために、 $i=i+1$ 、 $j=j+k$ とする（ステップS716）。次いで、所定のブロック数 r まで分割されたかどうかを判定するため、 i と r を比較する（ステップS7

18）。ステップS718の判定において、 $i=r$ であれば、まだ分割すべきブロックが残っているので、ステップS714に戻る。

【0151】ステップS718の判定において、 $i=r$ であれば、分割すべきブロックが残っていないので、 j が M の最終ビット番号未満かどうかを判定する（ステップS720）。 j が M の最終ビット番号未満であれば、 M の j ビット目から最後までを M_i として、データ番号5に格納して（ステップS722）次の処理へ移る。 j が M の最終ビット番号以上であれば、 $A||0$ をデータ番号5に格納して（ステップS724）次の処理へ移る。

【0152】図19は、暗号化処理の詳細を示すフローチャートであり、ステップS730以下の詳細を示す。まず、暗号化処理の制御変数 i の初期設定、 $i=0$ を行う（ステップS732）。制御変数 i は、分割された平文ブロックの番号を示すものである。次いで、データ番号 $(6+i)$ から平文ブロック M_i を読み出し、これにイデアル (n) を法とする e 乗法を行い、その結果を暗号化ブロック C_i として、データ番号 $(6+r+i)$ に格納する（ステップS734）。次いで、制御変数 i を1だけ増加させる（ステップS736）。

【0153】次いで、所定のブロック数 r まで暗号化されたかどうかを判定するため、 i と r とを比較する（ステップS738）。ステップS738の判定において、 $i=r$ であれば、まだ暗号化すべきメッセージブロックが残っているので、ステップS734に戻る。ステップS738の判定において、 $i=r$ であれば、暗号化が終了したので、次の処理へ移る。

【0154】図20は、暗号文送信処理の詳細を示すフローチャートであり、ステップS750以下の詳細を示す。まず、暗号文送信処理の制御変数 i の初期設定、 $i=0$ を行う（ステップS752）。制御変数 i は、暗号化ブロックの番号を示すものである。次いで、データ番号 $(6+r+i)$ から暗号化ブロック C_i を読み出し、送信する（ステップS754）。次いで、制御変数 i を1だけ増加させる（ステップS756）。

【0155】次いで、所定のブロック数 r まで送信されたかどうかを判定するため、 i と r とを比較する（ステップS758）。ステップS758の判定において、 $i=r$ であれば、まだ送信すべき暗号化ブロックが残っているので、ステップS754に戻る。ステップS758の判定において、 $i=r$ であれば、送信が終了したので、データ番号5の M を読み出し、 $M=A||0$ かどうかを判定する（ステップS760）。

【0156】ステップS760の判定において、 $A||0$ でなければ、まだ分割以下の処理を行うべき平文が残っているので、平文分割処理（ステップS710）に戻る。ステップS760の判定において、 $A||0$ であれば、暗号化送信処理が終了する。

【0-1-57】図2-1ないし図2-3は、受信復号化処理の詳細手順を示すフローチャートである。

【0-1-58】受信復号化処理では、表4に示すようなメモリ上のデータ配置を行う。

【0-1-59】

【表4】

データ番号	データ内容
1	公開鍵1 (n)
2	秘密鍵2 d
3	ブロック数 r
4	分割平文長 (1 ≤ 2n を超えない最大整数)
5	暗号文ブロックC0
...	...
5+r	暗号文ブロックCr-1
5+r-1	平文ブロックM0
...	...
5+2r	平文ブロックMr-1
5+2r+1	結合平文M

そして、図2-1に示すように、まずステップS801で、復号化のための初期設定が行われる。すなわちステップS801では、公開鍵1 (n)、秘密鍵2 d、ブロック数 r、分割平文長 (1 ≤ 2n を超えない最大整数) をそれぞれデータ番号1～4に設定する。次いで、暗号文ブロックC0～Cr-1 をブロック毎にデータ番号5～(5+r-1)へブロック毎に読み込む(ステップS803)。次いで、暗号文ブロック毎の復号化処理(ステップS810)、続いて平文統合処理(ステップS820)を行う。

【0-1-60】図2-2は、復号化処理の詳細を示すフローチャートであり、ステップS810以下の詳細を示す。まず、復号化処理の制御変数 i の初期設定、i = 0を行う(ステップS812)。制御変数 i は、暗号化ブロックの番号を示すものである。次いで、データ番号(5+i)から暗号化ブロックCiを読み出し、これにイデアル(n)を法とするd乗演算を行い、その結果を復号化された平文ブロックMiとして、データ番号(5+r+i)に格納する(ステップS814)。次いで、制御変数 i を1だけ増加させる(ステップS816)。

【0-1-61】次いで、所定のブロック数 r まで復号化されたか否かを判定するため、i と r とを比較する(ステップS818)。ステップS818の判定において、i

$$(C_0, C_1) = (123, 456) : 127 = (164, 1004) \bmod (1189) \cdots (46)$$

式(4-6)によって暗号化を行ない、受け手に送る。

【0-1-69】【復号化処理】受け手は自らの秘密鍵 d =

$$(164, 1004) : 463 = (123, 456) = (M_0, M_1) \bmod (1189) \cdots (47)$$

式(4-7)を計算することによって、復号化を行なう。

≠ r であれば、まだ復号化すべきブロックが残っているため、ステップS814へ戻る。ステップS818の判定において、i = r であれば、復号化が終了したので、次の平文統合処理へ移る。

【0-1-62】図2-3は、平文統合処理の詳細を示すフローチャートであり、ステップS820以下の詳細を示す。まず、平文統合処理の制御変数 i、j、k の初期設定を行う(ステップS822)。制御変数 i は、統合すべき平文ブロックの番号を示すものである。制御変数 j は、統合後の平文データ M を構成する各ビットのビット番号である。k は、ビット数で表した分割平文長である。ステップS822では、それぞれ、i = 0、j = 0、k = (1 ≤ 2n を超えない最大整数) と設定する。

【0-1-63】次いで、データ番号(5+r+1)からM1を読み出し、作業領域の j ビット目から(j+k-1)ビット目までに格納する(ステップS824)。次いで、制御変数更新のために、i = i + 1、j = j + k とする(ステップS826)。次いで、所定のブロック数 r まで統合されたか否かを判定するため、i と r とを比較する(ステップS828)。ステップS828の判定において、i ≠ r であれば、まだ統合すべきブロックが残っているため、ステップS824へ戻る。

【0-1-64】ステップS828の判定において、i = r であれば、統合すべきブロックが残っていないので、作業領域内に連続された r 個のブロックをデータ番号(5+2r)の末尾に連続して格納して(ステップS830)、受信復号化処理を終了する。

【0-1-65】【3次元分体(アイゼンシュタイン体)を使用した公開鍵暗号方式の数値例】次に、3次元分体(アイゼンシュタイン体)を使用した本発明に係る公開鍵暗号方式の実施の形態を具体的な数値例を用いて説明する。

【0-1-66】【鍵生成処理】2個の素イデアル P = (2-9)、Q = (4-1) (秘密鍵1) を生成し、その積 N = (1-8-9) の剰余類(公開鍵1)を決定する。次に秘密鍵1から L = 1-6-8-0 を計算し、e = 1-2-7 (公開鍵2) と、d = 4-6-3 (秘密鍵2) を生成する。

【0-1-67】【平文分割処理】平文 M = 1-2-3-4-5-6 に対して、剰余類の範囲内の分割平文 (M0, M1) = (1-2-3, 4-5-6) を生成する。

【0-1-68】【暗号化処理】受け手の公開鍵 e = 1-2-7 を用い、

【数4-4】

4-6-3 を用い、

【数4-5】

【0-1-70】【平文統合処理】復号化された分割平文

【0171】(M0, M1) = (123, 456) を接続し、原平文 M = 123456へ復元する。

【0171】【2次体(ガウス体)を利用した公開鍵暗号方式の数値例】次に、2次体(ガウス体)を利用した本発明に係る公開鍵暗号方式の実施形態を具体的な数値例を用いて説明する。

【0172】【鍵生成処理】秘密鍵1である2個の素イデアル P = (3, 1)、Q = (4, 7) を生成し、その積 N = (1, 457) の剰余類(公開鍵1)を決定する。

【0173】次いで、秘密鍵1から L = 22080 を計算し、(C0, C1) = (1, 23, 45, 6) 127
= (2, 46, 54, 5) mod (1457) ... (48)

式(48)によって暗号化を行い、暗号文(C0, C1) = (2, 46, 54, 5)を受け手に送る。

【0176】【復号化処理】受け手は自らの秘密鍵 d = (2, 46, 54, 5) 20836 = (M0, M1) mod (1457) ... (49)

式(49)を計算することによって、暗号文の復号化を行い、分割された平文(M0, M1) = (123, 456)を得る。

【0177】【平文統合処理】復号化された分割平文(M0, M1) = (123, 456) を接続し、原平文 M = 123456へ復元する。

【0178】次に、本発明に係る公開鍵暗号方式を利用した認証方式及び認証装置について説明する。図2は、本発明に係る公開鍵暗号方式による認証通信装置111の全体構成を示すブロック図である。

【0179】本発明の公開鍵暗号方式を用い、認証を行いたい者が、自らの秘密鍵により認証文の暗号化を行なって生成した認証子を受け手に送る方式による認証装置を構成することができる。

【0180】図2によれば、認証通信装置111は、鍵生成装置21と、認証文生成装置131と、認証文検証装置141と、通信路51とを含んで構成される。鍵生成装置21と、通信路51とは、図1の暗号化通信装置11に使用したものと同一であるのでその説明は省略する。

【0181】【認証用鍵生成処理】鍵生成装置21は、2個の素イデアル P、Q (秘密鍵1) を生成し、その積 N = P・Q の剰余類(公開鍵1)を決定する。次に素イデアル P、Q から L を計算し、e (公開鍵2) と、d (秘密鍵2) を生成する。

【0182】次いで、2個の素イデアル(秘密鍵1)の生成および剰余類(公開鍵1)の決定を行う。円分体の場合は、原始根の位数 m を入力として、秘密鍵1である素イデアル P、Q を出力する。2次体の場合は、判別式 D を入力として、秘密鍵1である素イデアル P、Q を出

$$(h(C)0, h(C)1, \dots, h(C)r-1) \\ = (h(M)0, h(M)1, \dots, h(M)r-1) d \mod N \dots (50)$$

式(50)によって認証子の暗号化を行なう。

【0188】以上より得られた暗号化認証子 h(C) =

算し、e = 127 (公開鍵2) と、d = 20836 (秘密鍵2) を生成する。

【0174】【平文分割処理】平文 M = 123456 に対して、それぞれ剰余類の範囲内の大きさのブロックに分割した分割平文(M0, M1) = (123, 456) を生成する。

【0175】【暗号化処理】受け手の公開鍵 e = 127 を用いて、【数46】

20836を用いて、【数47】

力する。

【0183】次いで、公開鍵2である e と秘密鍵2である d の生成を行う。円分体、2次体ともに、秘密鍵1である P、Q を入力として、公開鍵2である e と秘密鍵2である d を出力する。また、鍵生成装置21は、ブロック数 r、及び分割平文長 [log2 n] を認証文生成装置131へ出力する。以上の鍵生成処理は、図1の暗号化通信装置11における鍵生成装置21の動作と同じである。

【0184】【認証文生成処理】認証文生成装置131は、認証文を受け入れる認証文入力部133と、認証文をハッシュ化して認証子を生成する認証文ハッシュ化処理部135と、認証子をブロックに分割する認証子分割処理部137と、分割認証子を暗号化する認証子暗号化処理部139とを含んで構成されている。

【0185】認証文入力部133により受け入れられた認証文 M は、認証文ハッシュ化処理部135により、ハッシュ関数 h を用いてハッシュ化され、認証子 h(M) が生成される。ハッシュ関数 h は特に限定されるものではない。

【0186】次いで、認証子 h(M) は、認証子分割処理部137により、それぞれ分割平文長 [log2 n] の長さの r 個のブロックずつに分割される。ここで、r を Z 上の整数環 O の次元とする。認証子 h(M) に対して、剰余類の範囲内の分割認証子 h(M) = (h(M)0, h(M)1, ..., h(M)r-1) を生成する。

【0187】次いで、認証子暗号化処理部139により、送り手の秘密鍵 d を用いて、【数48】

(h(C)0, h(C)1, ..., h(C)r-1) と認証文 M の組を通信路51を介して、受け手すなわち認証

文検証装置1.4.1に送る。

【0.1.89】認証文検証装置1.4.1は、認証子復号化処理部1.4.3と、認証子統合処理部1.4.5と、認証子ハッシュ化処理部1.4.7と、認証子照合部1.4.9とを含んで構成されている。

【0.1.90】認証文検証装置1.4.1は、認証文生成装置1.3.1から暗号化認証子 $h(C) = (h(C)_0, h(C)_1, \dots, h(C)_{r-1})_e$

$$\equiv (h_0(M)_0, h_0(M)_1, \dots, h_0(M)_{r-1}) \pmod{N} \quad (S1)$$

式(5.1)を計算することによって、暗号化認証子を復号化して分割認証子を得る。

【0.1.92】次いで、認証子統合処理部1.4.5は、復号化された分割認証子 $(h_0(M)_0, h_0(M)_1, \dots, h_0(M)_{r-1})$ を統合し、認証子 $h(M)$ を生成する。一方、認証文ハッシュ化処理部1.4.7は、送られて来た認証文 M をハッシュ関数 h により、ハッシュ化して認証子 $h(M)^*$ とする。

【0.1.93】次いで、認証子照合部1.4.9により、認証

$(C)_1, \dots, h(C)_{r-1})$ と認証文 M の組を受け取る。また、鍵生成装置2.1から通信路5.1を介して、公開鍵 $1N$ 、および公開鍵 $2e$ を受け取る。

【0.1.91】次いで、認証子復号化処理部1.4.3により、暗号化認証子を復号化する。すなわち、認証子復号化処理部1.4.3は、送り手の公開鍵 e を利用して、

【数4.9】

子 $h(M)$ と $h(M)^*$ とを比較し、一致していれば認証文が正当であり、不一致であれば認証文が不当であるとして、認証文の正当性を判断する。

【0.1.94】図2.4ないし図2.7は、認証文生成処理の詳細手順を示すフローチャートである。認証文生成処理では、表5に示すようなメモリ上のデータ配置を行う。

【0.1.95】

【表5】

データ番号	データ内容
1	公開鍵 $1(n)$
2	秘密鍵 $2(d)$
3	ブロック数 r
4	分割平文長 $(1 \leq n \leq n)$
5	認証文 M
6	認証子 $h(M)$
7	認証子ブロック $h(M)_i(i)$
...	...
$6+r-1$	認証子ブロック $h(M)_{r-1}$
$6+r$	暗号化認証子ブロック $h(C)_0$
...	...
$6+2r$	暗号化認証子ブロック $h(C)_{r-1}$

そして、図2.4に示すように、まずステップS.9.0.1で、認証文生成のための初期設定が行われる。すなわちステップS.9.0.1では、公開鍵 $1(n)$ 、秘密鍵 $2(d)$ 、ブロック数 r 、分割平文長 $(1 \leq n \leq 2n)$ を超えない最大整数)をそれぞれデータ番号1~4に設定する。次いで、認証文 M をデータ番号5に読み込む(ステップS.9.0.3)。

【0.1.96】次いで、認証文 M をデータ番号5から読み出し、ハッシュ関数 h でハッシュ化し、その結果である認証子 $h(M)$ をデータ番号6に格納する(ステップS.9.0.5)。

【0.1.97】次いで、認証子 $h(M)$ の先頭からそれぞれ分割平文長のビット数の長さのブロックを r 個切り出す認証子分割処理を行い(ステップS.9.2.0)、続いて、認証子暗号化処理を行う(ステップS.9.4.0)。暗

号化された認証子ブロックは、暗号送信処理され(ステップS.9.6.0)、また認証子が残っていれば、認証子分割処理(ステップS.9.2.0)に戻り、認証子が残っていないれば終了する。

【0.1.98】図2.5は、認証子分割処理の詳細を示すフローチャートであり、ステップS.9.2.0以下の詳細を示す。まず、認証子分割処理の制御変数 i, j, k の初期設定を行う(ステップS.9.2.2)。制御変数 i は、分割された認証子ブロック $h(M)_i(i)$ の番号 i を示すものである。制御変数 j は、分割前の認証子 $h(M)$ を構成する各ビットのビット番号である。 k は、ビット数で表した分割平文長である。ステップS.9.2.2では、それぞれ、 $i=0, j=0, k=(1 \leq n \leq 2n)$ を超えない最大整数)と設定する。

【0.1.99】次いで、データ番号6から認証子 $h_0(M)$

を読み出し、 $h_i(M)$ の j ビット目から $(j+k-1)$ ビット目までを切り出し、これを $h_i(M)_i$ としてデータ番号 $(7+i)$ に格納する (ステップ S924)。次いで、制御変数更新のために、 $i=i+1$ 、 $j=j+k$ とする (ステップ S926)。次いで、所定のブロック数 r まで分割されたか否かを判定するため、 i と r とを比較する (ステップ S928)。ステップ S928 の判定において、 $i \neq r$ であれば、まだ分割すべきブロックが残っているので、ステップ S924 へ戻る。

【0200】ステップ S928 の判定において、 $i = r$ であれば、分割すべきブロックが残っていないので、 j が $h_i(M)$ の最終ビット番号未満が否かを判定する (ステップ S930)。 j が $h_i(M)$ の最終ビット番号未満であれば、 $h_i(M)$ の j ビット目から最後までを $h_i(M)_i$ として、データ番号 6 に格納して (ステップ S932)、次の認証子暗号化処理へ移る。 j が $h_i(M)$ の最終ビット番号以上であれば、 $A_i = 1$ 、 $O_i = 0$ をデータ番号 6 に格納して (ステップ S934)、次の認証子暗号化処理へ移る。

【0201】図 26 は、認証子暗号化処理の詳細を示すフローチャートであり、ステップ S940 以下の詳細を示す。まず、認証子暗号化処理の制御変数 i の初期設定、 $i=0$ を行う (ステップ S942)。制御変数 i は、分割された認証子ブロックの番号を示すものである。次いで、データ番号 $(7+i)$ から認証子ブロック $h_i(M)_i$ を読み出し、これにイデアル (n) を法とする d 乗演算を行い、その結果を暗号化認証子ブロック $h_i(C)_i$ として、データ番号 $(7+r+i)$ に格納する (ステップ S944)。次いで、制御変数 i を 1 だけ増加させる (ステップ S946)。

【0202】次いで、所定のブロック数 r まで暗号化されたか否かを判定するため、 i と r とを比較する (ステ

ップ S948)。ステップ S948 の判定において、 $i \neq r$ であれば、まだ暗号化すべき認証子ブロックが残っているので、ステップ S944 へ戻る。ステップ S948 の判定において、 $i = r$ であれば、暗号化が終了したので、次の暗号文送信処理へ移る。

【0203】図 27 は、暗号文送信処理の詳細を示すフローチャートであり、ステップ S960 以下の詳細を示す。まず、暗号文送信処理の制御変数 i の初期設定、 $i=0$ を行う (ステップ S962)。制御変数 i は、暗号化ブロックの番号を示すものである。次いで、データ番号 $(7+r+i)$ から暗号化認証子ブロック $h_i(C)_i$ を読み出し、送信する (ステップ S964)。次いで、制御変数 i を 1 だけ増加させる (ステップ S966)。

【0204】次いで、所定のブロック数 r まで送信されたか否かを判定するため、 i と r とを比較する (ステップ S968)。ステップ S968 の判定において、 $i \neq r$ であれば、まだ送信すべき暗号化ブロックが残っているので、ステップ S964 へ戻る。ステップ S968 の判定において、 $i = r$ であれば、送信が終了したので、データ番号 6 の $h_i(M)_i$ を読み出し、 $h_i(M)_i = A_i = 1$ 、 $O_i = 0$ が否かを判定する (ステップ S970)。

【0205】ステップ S970 の判定において、 $A_i = 1$ 、 $O_i = 0$ でなければ、まだ分割以下の処理を行うべき認証子が残っているので、認証子分割処理 (ステップ S920) へ戻る。ステップ S970 の判定において、 $A_i = 1$ 、 $O_i = 0$ であれば、認証文生成処理が終了する。

【0206】図 28 ないし図 31 は、認証文復号化処理の詳細手順を示すフローチャートである。

【0207】認証文復号化処理では、表 6 に示すようなメモリ上のデータ配置を行う。

【0208】

【表 6】

データ番号	データ内容
1	公開鍵 g, G
2	公開鍵 g, G
3	ブロック数 r
4	分割数 n 及び $1/n$ の逆元
5	認証文 A
6	暗号化認証子ブロック $h_i(C)_i$
7	...
8	暗号化認証子ブロック $h_i(C)_i$
9	...
10	...
11	...
12	...
13	...
14	...
15	...
16	...
17	...
18	...
19	...
20	...
21	...
22	...
23	...
24	...
25	...
26	...
27	...
28	...
29	...
30	...
31	...
32	...
33	...
34	...
35	...
36	...
37	...
38	...
39	...
40	...
41	...
42	...
43	...
44	...
45	...
46	...
47	...
48	...
49	...
50	...
51	...
52	...
53	...
54	...
55	...
56	...
57	...
58	...
59	...
60	...
61	...
62	...
63	...
64	...
65	...
66	...
67	...
68	...
69	...
70	...
71	...
72	...
73	...
74	...
75	...
76	...
77	...
78	...
79	...
80	...
81	...
82	...
83	...
84	...
85	...
86	...
87	...
88	...
89	...
90	...
91	...
92	...
93	...
94	...
95	...
96	...
97	...
98	...
99	...
100	...

そして、認証文復号化処理では、まず図 28 に示すよう

に、ステップ S1000 で、復号化のための初期設定が行

われる。すなわちステップS10001では、公開鍵 e 、公開鍵 $2e$ 、ブロック数 r 、分割平文長 $(\log_2 n)$ を超えない最大整数)をそれぞれデータ番号1～4に設定する。

【0209】次いで、認証文 M をデータ番号5へ、暗号化認証子ブロック $h(C)$ 、 $0 \sim h(C)-1$ をブロック毎にデータ番号6～ $(5+r)$ へブロック毎に読み込む(ステップS1003)。次いで、ブロック毎の認証子復号化処理(ステップS1020)、続いて認証子統合処理(ステップS1040)、認証文ハッシュ化処理(ステップS1060)、認証確認処理(ステップS1080)を順次行う。

【0210】図29は、認証子復号化処理の詳細を示すフローチャートであり、ステップS1020以下の詳細を示す。まず、認証子復号化処理の制御変数 i の初期設定、 $i=0$ を行う(ステップS1022)。制御変数 i は、暗号化認証子ブロックの番号を示すものである。次いで、データ番号 $(6+i)$ から暗号化認証子ブロック $h(C)$ 、 i を読み出し、これにイデアル (n) を法とする e 乗演算を行い、その結果を復号化された認証子ブロック $h(M)$ 、 i として、データ番号 $(6+r+i)$ に格納する(ステップS1024)。次いで、制御変数 i を1だけ増加させる(ステップS1026)。

【0211】次いで、所定のブロック数 r まで復号化されたか否かを判定するため、 i と r とを比較する(ステップS1028)。ステップS1028の判定において、 $i \neq r$ であれば、まだ復号化すべきブロックが残っているので、ステップS1024へ戻る。ステップS1028の判定において、 $i = r$ であれば、復号化が終了したので、次の認証子統合処理へ移る。

【0212】図30は、認証子統合処理の詳細を示すフローチャートであり、ステップS1040以下の詳細を示す。まず、認証子統合処理の制御変数 i 、 j 、 k の初期設定を行う(ステップS1042)。制御変数 i は、統合すべき認証子ブロック $h(M)$ 、 i の番号 i を示すものである。制御変数 j は、統合後の認証子データ $h(M)$ を構成する各ビットのビット番号である。 k は、ビット数で表した分割平文長である。ステップS1042では、それぞれ、 $i=0$ 、 $j=0$ 、 $k=(\log_2 n)$ を超えない最大整数)と設定する。

【0213】次いで、データ番号 $(6+r+i)$ から $h(M)$ 、 i を読み出し、作業領域の j ビット目から $(j+k-1)$ ビット目までに格納する(ステップS1044)。次いで、制御変数更新のために、 $i=i+1$ 、 $j=j+k$ とする(ステップS1046)。

【0214】次いで、所定のブロック数 r まで統合されたか否かを判定するため、 i と r とを比較する(ステップS1048)。ステップS1048の判定において、 $i \neq r$ であれば、まだ統合すべきブロックが残っているので、ステップS1044へ戻る。

【0215】ステップS1048の判定において、 $i = r$ であれば、統合すべきブロックが残っていないので、作業領域内に接続して形成された統合認証子 $h(M)$ をデータ番号 $(6+2r)$ の末尾に接続して格納し(ステップS1050)、次の認証文ハッシュ化処理へ移る。

【0216】図31は、認証文ハッシュ化処理および認証確認処理の詳細を示すフローチャートであり、ステップS1060以下の詳細を示す。認証文ハッシュ化処理は、データ番号5から認証文 M を読み出し、ハッシュ関数 h でハッシュ化し、その結果を $h(M)^*$ として、データ番号 $(7+2r)$ に格納し(ステップS1062)、次の認証確認処理へ移る。

【0217】認証確認処理は、データ番号 $(6+2r)$ の統合認証子 $h(M)$ とデータ番号 $(7+2r)$ のハッシュ化認証子 $h(M)^*$ とを比較する(ステップS1082)。

【0218】この比較の結果、 $h(M)$ と $h(M)^*$ とが一致していれば、認証成功を出力して(ステップS1084)終了し、 $h(M)$ と $h(M)^*$ とが一致しなければ、認証失敗を出力して(ステップS1086)終了する。

【0219】なお、この認証通信の実施形態において、暗号化認証子 $h(C)$ と認証文 M の組をさらに受信者の公開鍵を用いて暗号化する暗号化認証通信も、認証通信と暗号化通信とを組み合わせることで従来の公開鍵暗号と同様に実現できる。

【0220】

【発明の効果】以上説明したように、本発明に係る公開鍵暗号方式によれば、従来のRSA暗号方式と比較して、完全読解についての安全性は同等以上の強度が得られるという効果を奏する。

【0221】また、本発明の公開鍵暗号方式に対して、従来の同報通信攻撃法は無効となるので、同報通信攻撃に対して著しく強度が増した暗号方式を提供することができるという効果を奏する。

【0222】また、本発明の公開鍵暗号方式によれば、暗号化にかかる計算時間の大部分を占める乗算回数が、楕円曲線上のRSA暗号と比較して2/5に削減され、大幅な高速化を実現することができるという効果を奏する。

【0223】また、本発明の拡大体への拡大次数が2次の場合は、従来のRSA暗号と比較して同程度の暗号化速度を確保しつつ、強度を高めることができるという効果を奏する。

【0224】また、本社から植数支社への同報データ通信や多地点間でのテレビ会議、有線あるいは無線電話などによる連絡などにおいて、守秘性を高めたいときに、本発明による暗号方式を用いてデータ、画像、音声の暗号化を行なうことにより、従来の方式に比べて同報通信攻撃に対して強度の増した通信を行なうことができると

いう効果を奏する。

【0225】また、本発明をインターネットなどのネットワークにおいて用いられているセキュリティ電子メールなどの暗号通信に適用すれば、同報通信攻撃に対する強度を増すことができるという効果を奏する。

【0226】また、本発明に係る認証装置は、そのハードウェア資源ならびにソフトウェア資源のそれぞれ大部分を暗号装置と共用することができ、暗号化および認証の双方に对称使用が可能となるという効果を奏する。

【0227】また、本暗号装置および認証装置を使用することにより、同報通信攻撃に強いネットワーク管理プロトコルを設計することができるという効果を奏する。

【0228】また、ネットワークを利用して行なわれる電子商取引や電子金融取引等において、本暗号装置および認証装置を使用することにより、取引データの暗号化や取引相手の相互認証におけるセキュリティを向上させることができるという効果を奏する。

【0229】また、本発明の暗号化鍵生成方法および暗号化鍵生成装置によれば、代数体上の整数環における素イデアルを利用することにより、従来の有理整数環上の素数に比べて利用可能な暗号化鍵を増加させることができるという効果を奏する。

【図面の簡単な説明】

【図1】本発明に係る公開鍵暗号方式を用いた暗号化通信装置の概要を示すシステム構成図である。

【図2】本発明に係る公開鍵暗号方式による認証通信装置の概要を示すシステム構成図である。

【図3】円分体の素イデアル(秘密鍵1)生成処理部。

【図4】2次体の素イデアル(秘密鍵1)生成処理部。

【図5】円分体の公開鍵2、秘密鍵2の生成処理部。

【図6】平分分割処理部の概略フローチャートである。

【図7】イデアル n を法とする乗算器の構成を示すブロック図である。

【図8】イデアル n を法とする2乗演算器の構成を示すブロック図である。

【図9】円分体暗号における鍵生成処理のフローチャート図(1/4)であり、円分体暗号における鍵生成のメインルーチンを示す。

【図10】円分体暗号における鍵生成処理のフローチャート図(2/4)であり、オイラー関数値 $\phi(m)$ の計算ルーチンを示す。

【図11】円分体暗号における鍵生成処理のフローチャート図(3/4)であり、素イデアル1、 p の生成ルーチンを示す。

【図12】円分体暗号における鍵生成処理のフローチャート図(4/4)であり、素イデアル2、 q の生成ルーチンを示す。

【図13】2次体暗号における鍵生成処理のフローチャート図(1/4)であり、2次体暗号における鍵生成のメインルーチンを示す。

【図14】2次体暗号における鍵生成処理のフローチャート図(2/4)であり、判別式 D の値を格納するルーチンを示す。

【図15】2次体暗号における鍵生成処理のフローチャート図(3/4)であり、素イデアル1、 p の生成ルーチンを示す。

【図16】2次体暗号における鍵生成処理のフローチャート図(4/4)であり、素イデアル2、 q の生成ルーチンを示す。

【図17】暗号化送信処理のフローチャート図(1/4)であり、暗号化送信処理のメインルーチンを示す。

【図18】暗号化送信処理のフローチャート図(2/4)であり、平文分割処理ルーチンを示す。

【図19】暗号化送信処理のフローチャート図(3/4)であり、暗号化処理ルーチンを示す。

【図20】暗号化送信処理のフローチャート図(4/4)であり、暗号文送信処理ルーチンを示す。

【図21】受信復号化処理のフローチャート図(1/3)であり、受信復号化処理のメインルーチンを示す。

【図22】受信復号化処理のフローチャート図(2/3)であり、復号化処理ルーチンを示す。

【図23】受信復号化処理のフローチャート図(3/3)であり、平文統合処理ルーチンを示す。

【図24】認証文生成処理のフローチャート図(1/4)であり、認証文生成処理のメインルーチンを示す。

【図25】認証文生成処理のフローチャート図(2/4)であり、認証子分割処理ルーチンを示す。

【図26】認証文生成処理のフローチャート図(3/4)であり、認証子暗号化ルーチンを示す。

【図27】認証文生成処理のフローチャート図(4/4)であり、暗号文送信処理ルーチンを示す。

【図28】認証文復号化処理のフローチャート図(1/4)であり、認証文復号化処理のメインルーチンを示す。

【図29】認証文復号化処理のフローチャート図(2/4)であり、認証子復号化処理ルーチンを示す。

【図30】認証文復号化処理のフローチャート図(3/4)であり、認証子統合処理ルーチンを示す。

【図31】認証文復号化処理のフローチャート図(4/4)であり、認証子ハッシュ化処理ルーチンおよび認証確認処理ルーチンを示す。

【図32】情性する素数の積で生成されるイデアル (n) による剰余類を2次元平面上に示した図である。

【図33】従来のRSA暗号方式の概要を示すシステム構成図である。

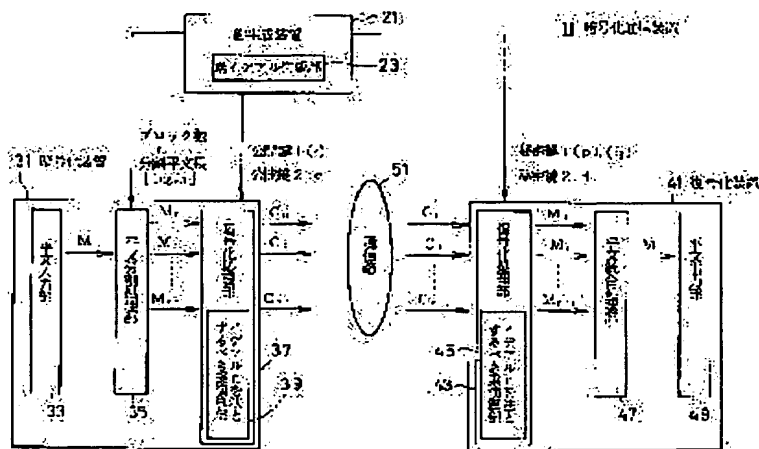
【符号の説明】

1…暗号化通信装置、2…鍵生成装置、23…素イデアル生成部、31…暗号化装置、33…平文入力部、35…平文分割部、37…暗号化処理部、39…イデアル n を法とするべき乗演算部、41…復号化装置、43…

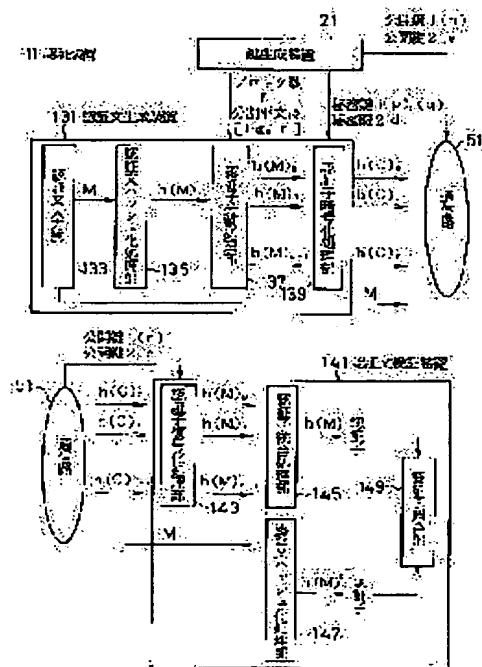
…復号化処理部、4.5…イデアル n を法とするべき乗演算部、4.7…平文統合処理部、4.9…平文出力部、5.1

…通信路、

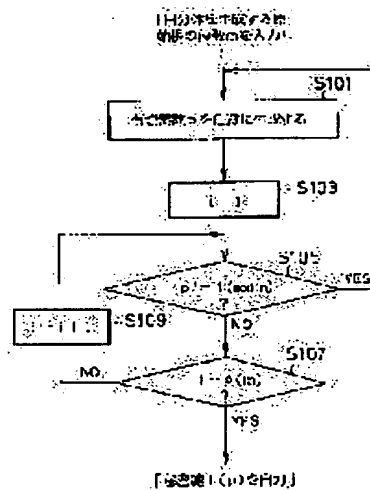
【図1】

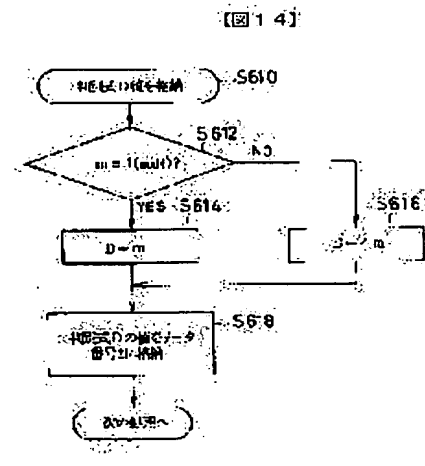
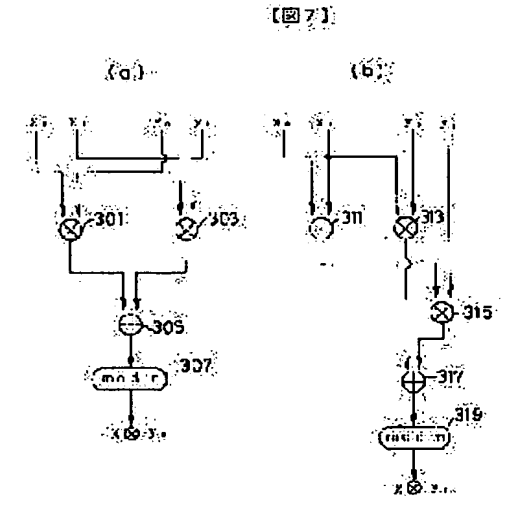
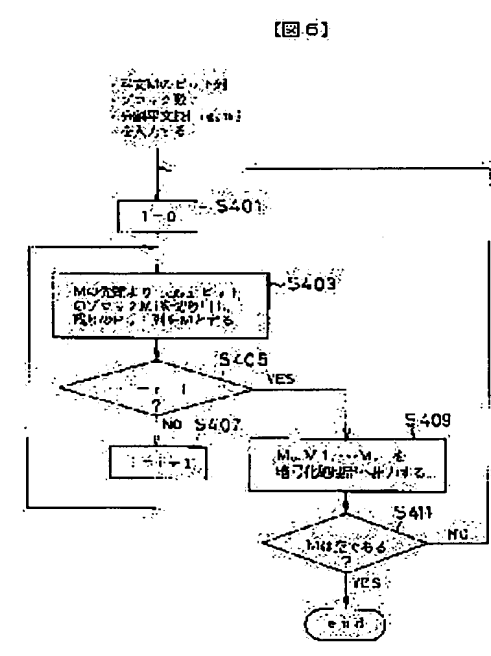
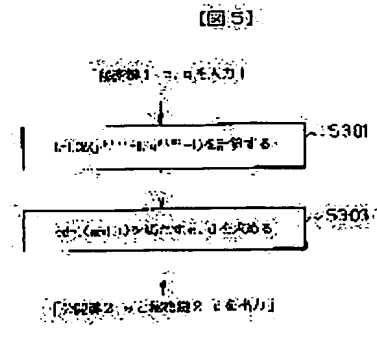
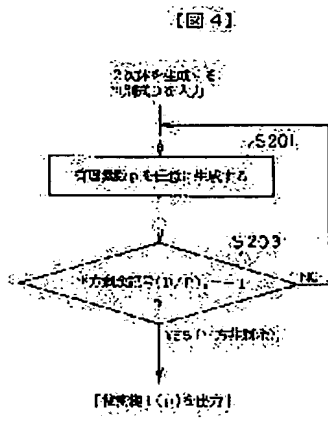


【図2】

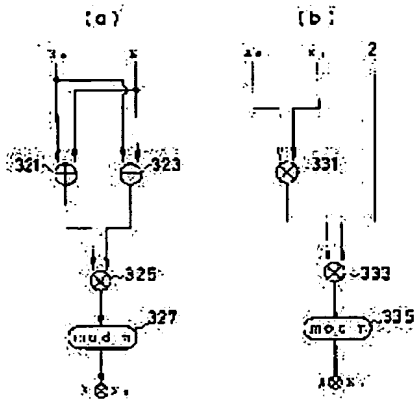


【図3】

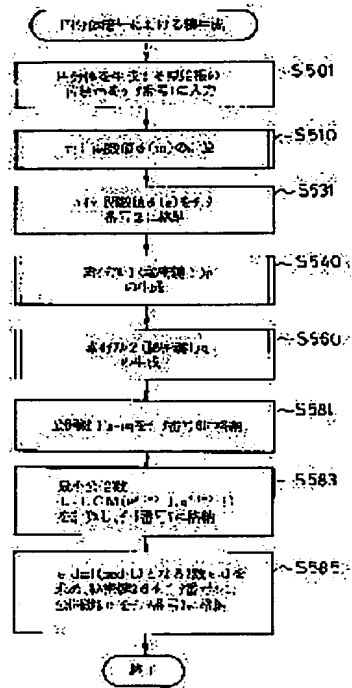




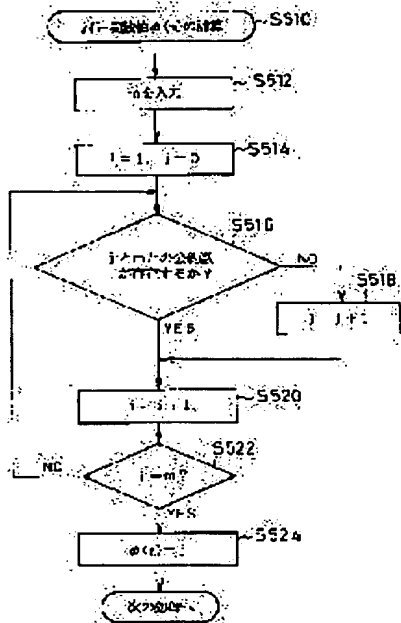
【図 8】



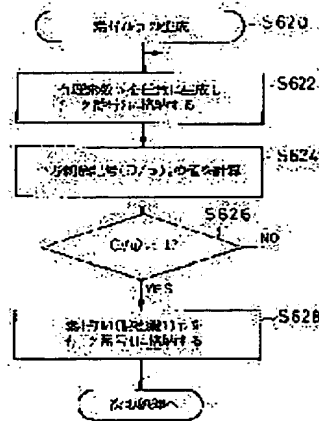
【図 9】



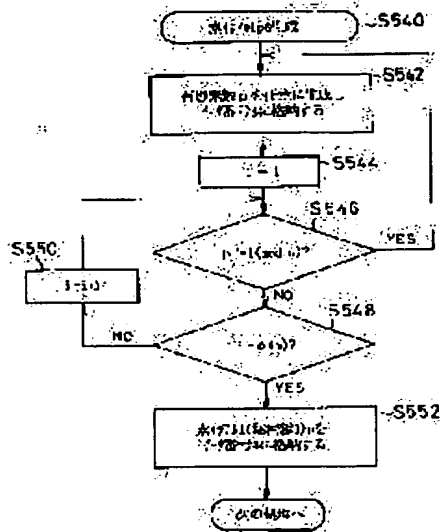
【図 10】



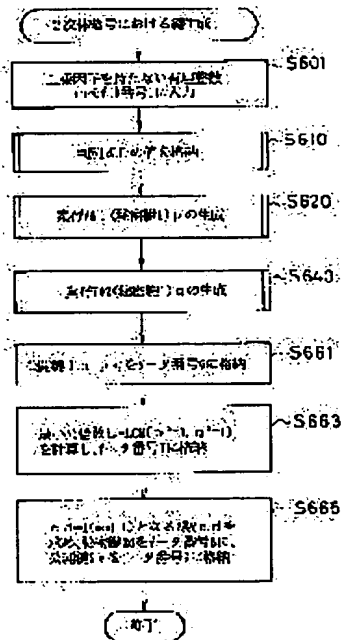
【図 11】



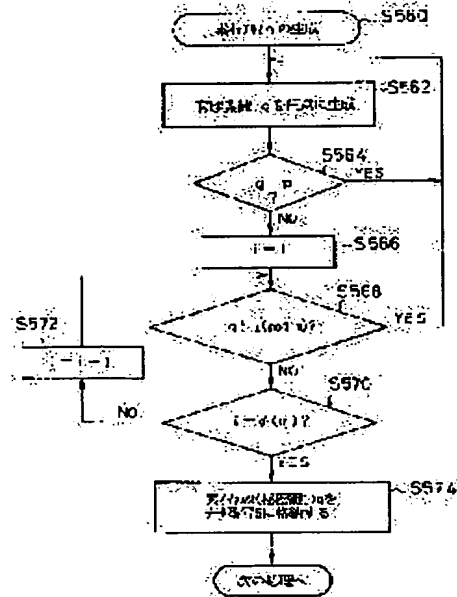
【図 1-1】



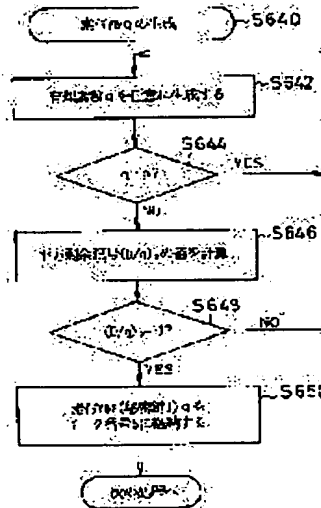
【図 1-3】



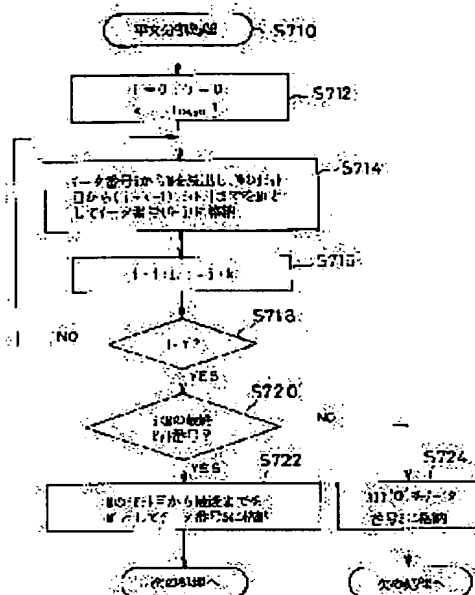
【図 1-2】



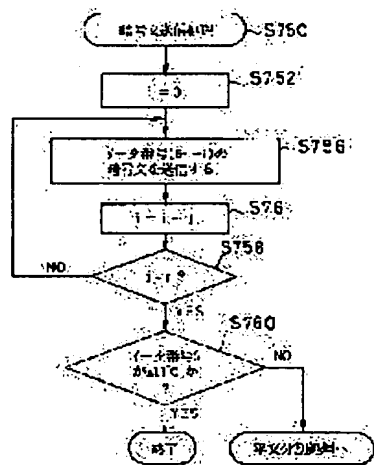
【図 1-6】



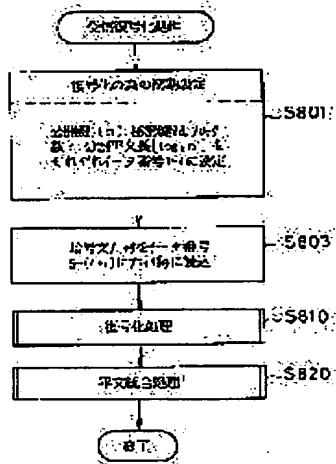
【圖 1-8】



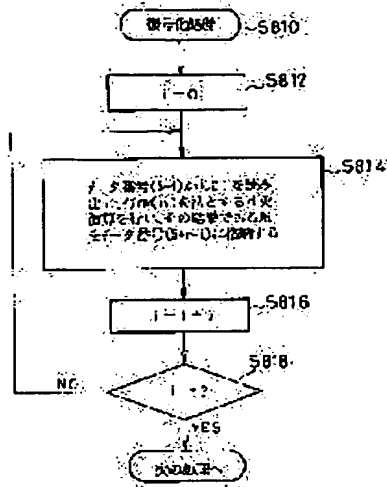
【圖 2.0】



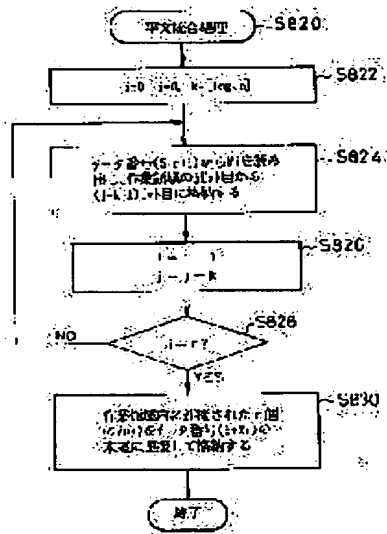
【図2-1】



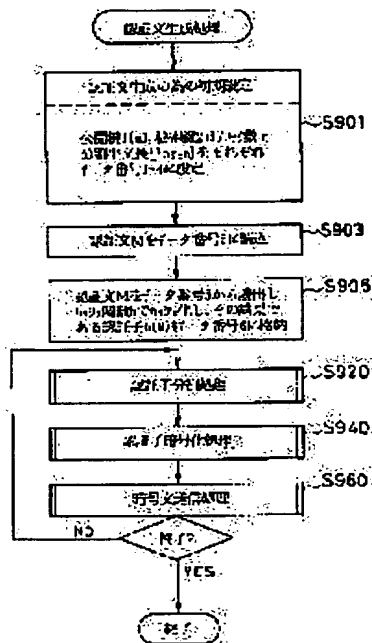
【図2-2】



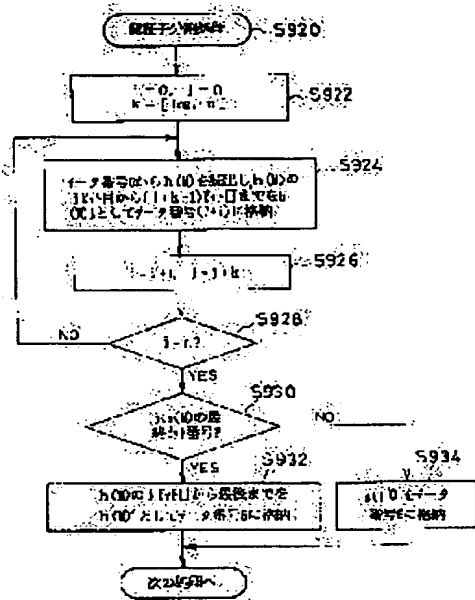
【図2-3】



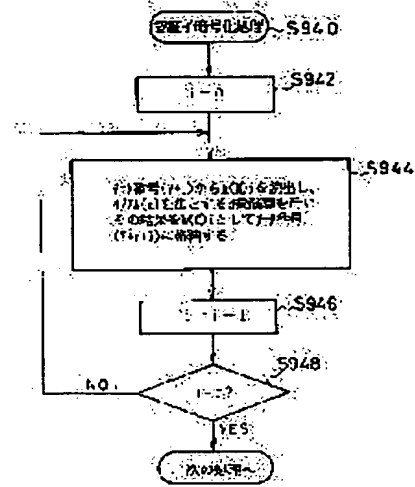
【図2-4】



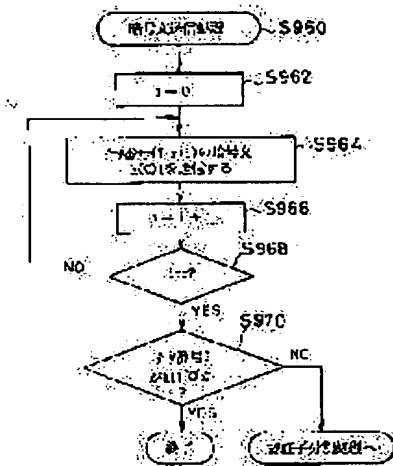
【図2.5】



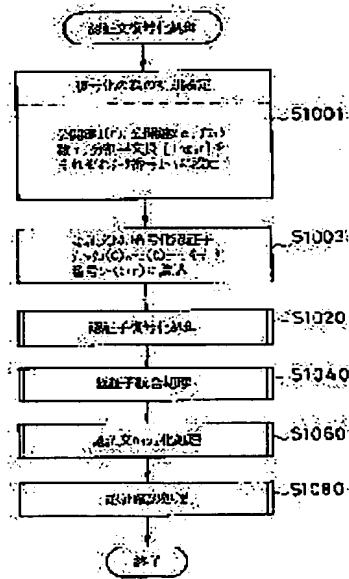
【図2.6】



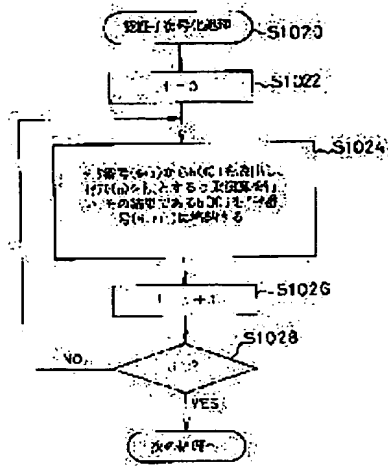
【図2.7】



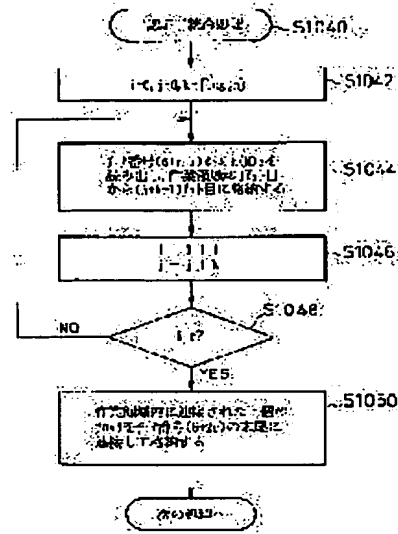
【図2.8】



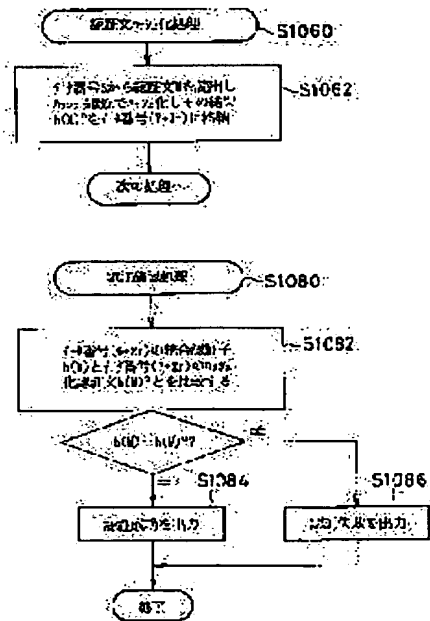
【図29】



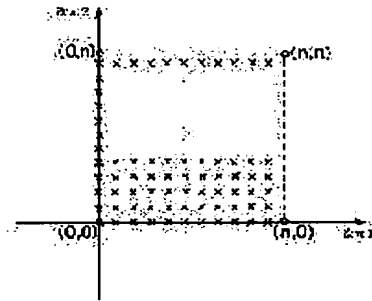
【図30】



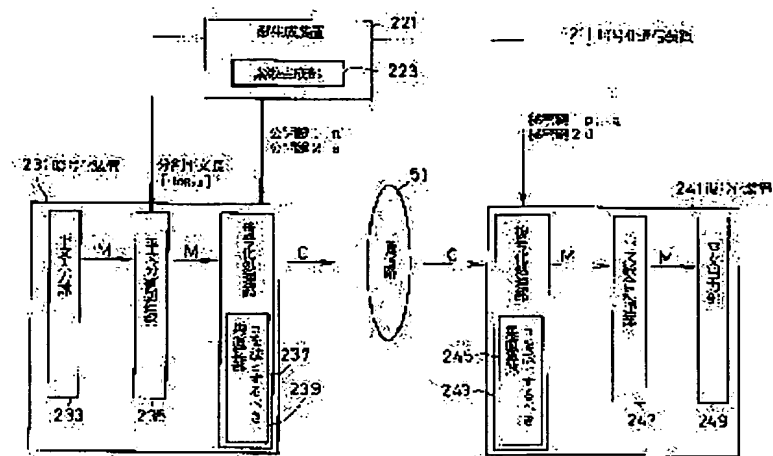
【図31】



【図32】



【図33】



フロントページの続き

(51) Int. Cl. 6

H 0 4 L 9/30
9/32

識別記号

庁内整理番号

F I

H 0 4 L 9/00

技術表示箇所

6 0 1 F
6 6 3 B
6 7 5 B

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☒ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☒ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☒ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.